U.S. DEPARTMENT OF COMMERCE
National Technical Information Service

PB-300 471

A Lagrangian Photochemical Air Quality Simulation Model: Adaptation to the St. Louis RAPS Data Base. Volume II. User's Manual

Environmental Research and Technology, Inc, Westlake Village, CA

Prepared for

Environmental Sciences Research Lab, Research Triangle Park, NC

Jun 79

United States
Environmental Protection
Agency

Environmental Sciences Research
Laboratory
Research Triangle Park NC 27711

EPA-600/8-79-015b
June 1979

Research and Development

PB 300471

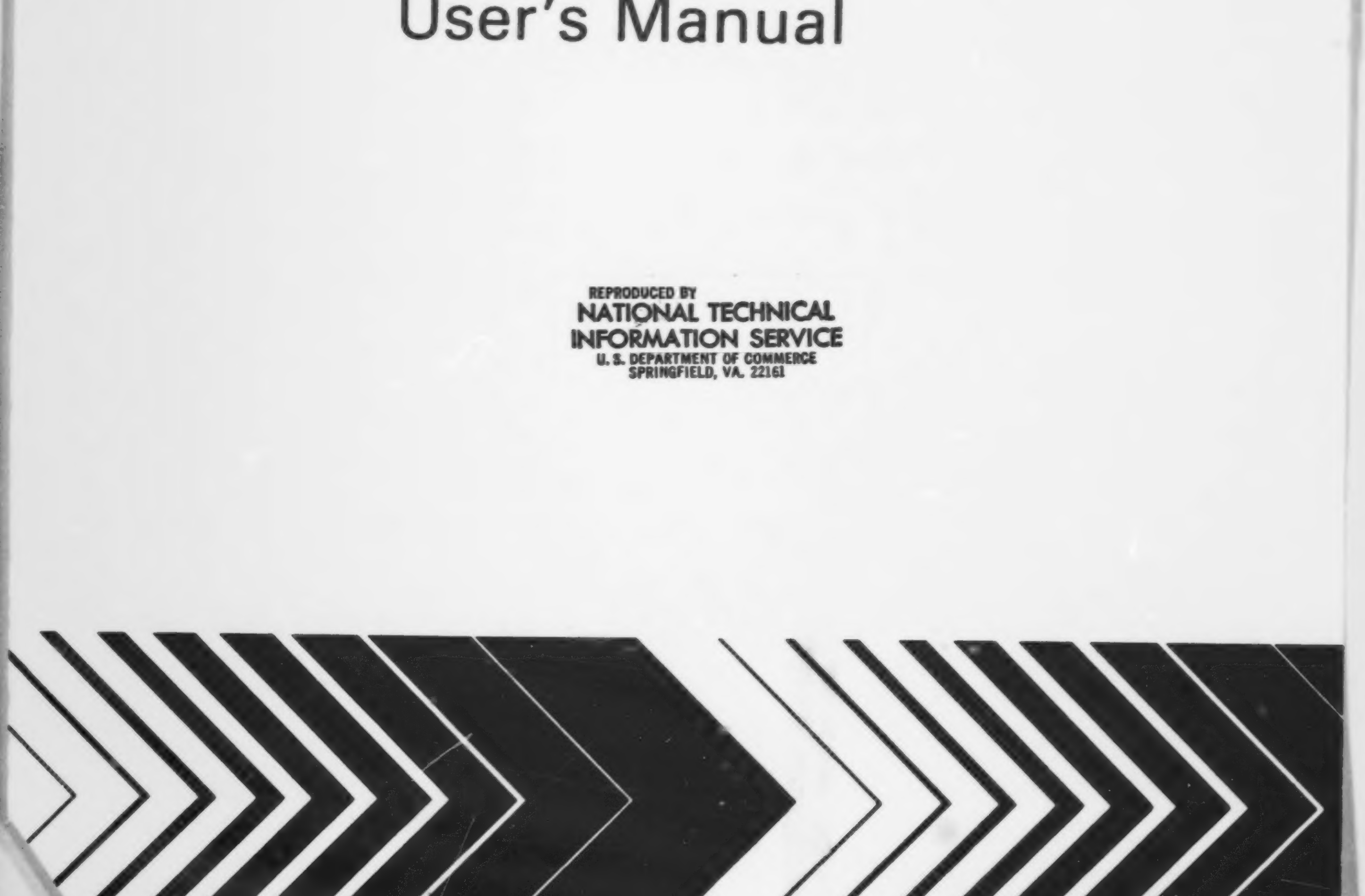
EPA

A Lagrangian Photochemical Air Quality Simulation Model

Adaptation to the St. Louis—RAPS Data Base

Volume II. User's Manual

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U. S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA. 22161



RESEARCH REPORTING SERIES

Research reports of the Office of Research and Development, U.S. Environmental Protection Agency, have been grouped into nine series. These nine broad categories were established to facilitate further development and application of environmental technology. Elimination of traditional grouping was consciously planned to foster technology transfer and a maximum interface in related fields. The nine series are:

1. Environmental Health Effects Research
2. Environmental Protection Technology
3. Ecological Research
4. Environmental Monitoring
5. Socioeconomic Environmental Studies
6. Scientific and Technical Assessment Reports (STAR)
7. Interagency Energy-Environment Research and Development
8. "Special" Reports
9. Miscellaneous Reports

This report has been assigned to the SPECIAL REPORTS series. This series is reserved for reports which are intended to meet the technical information needs of specifically targeted user groups. Reports in this series include Problem Oriented Reports, Research Application Reports, and Executive Summary Documents. Typical of these reports include state-of-the-art analyses, technology assessments, reports on the results of major research and development efforts, design manuals, and user manuals.

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completing)

1. REPORT NO. EPA-600/8-79-015b	2.	3. RECIPIENT'S ACCESSION NO. PB300471
4. TITLE AND SUBTITLE A LAGRANGIAN PHOTOCHEMICAL AIR QUALITY SIMULATION MODEL Adaptation to the St. Louis - RAPS Data Base Volume II. User's Manual		5. REPORT DATE June 1979 6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) Fred Lurmann, Daniel Godden, Alan C. Lloyd and Richard A. Nordsieck		8. PERFORMING ORGANIZATION REPORT NO.
9. PERFORMING ORGANIZATION NAME AND ADDRESS Environmental Research and Technology, Inc. 2625 Townsgate Road Westlake Village, CA 91361		10. PROGRAM ELEMENT NO. 1AA603A AA-045 (FY- 79) 11. CONTRACT/GRANT NO. 68-02-2765
12. SPONSORING AGENCY NAME AND ADDRESS Environmental Sciences Research Laboratory - RTP, NC Office of Research and Development U.S. Environmental Protection Agency Research Triangle Park, NC 27711		13. TYPE OF REPORT AND PERIOD COVERED Final 14. SPONSORING AGENCY CODE EPA/600/09

15. SUPPLEMENTARY NOTES

Volume I. Model Formulation -- EPA-600/8-79-015a, June 1979

16. ABSTRACT

A set of instructions have been compiled for use of a Lagrangian photochemical air quality simulation model adapted to the St. Louis, Missouri/Illinois metropolitan region and the Regional Air Pollution Study (RAPS) data base. The computer model, developed by Environmental Research and Technology, Inc., consists of a set of computer programs for the simulation of atmospheric transport, turbulent diffusion, and chemical kinetics of photochemical pollutants. The model is used to predict atmospheric concentrations of ozone, nitrogen dioxide, carbon monoxide, sulfur dioxide, and sulfate within an air column moving at the mean wind speed.

Descriptions of the meteorological, source emissions, and air quality data requirements, as well as sample input and output files, are provided. The computational procedures for using the model and a listing of the computer code are included.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group
* Air pollution * Hydrocarbons * Nitrogen oxides * Ozone Data * Adaptation * Mathematical models * Photochemical reactions * Manuals		13B 07C 07B 12A 07E 05B

18. DISTRIBUTION STATEMENT RELEASE TO PUBLIC	19. SECURITY CLASS *(This Report)* UNCLASSIFIED	21. NO. OF PAGES 448
	20. SECURITY CLASS *(This page)* UNCLASSIFIED	22. PRICE PC A19 / MF A01

EPA Form 2220-1 (9-73)

EPA-600/8-79-015b
June 1979

A LAGRANGIAN PHOTOCHEMICAL AIR QUALITY SIMULATION MODEL

Adaptation to the St. Louis - RAPS Data Base

Volume II. User's Manual

by

Fred Lurmann, Daniel Godden, Alan C. Lloyd, Richard A. Nordsieck
Environmental Research and Technology, Inc.
Environmental Analysis Division
2625 Townsgate Road
Westlake Village, California 91361

Contract No. 68-02-2765

Project Officer

Jack H. Shreffler
Meteorology and Assessment Division
Environmental Sciences Research Laboratory
Research Triangle Park, NC 27711

ENVIRONMENTAL SCIENCES RESEARCH LABORATORY
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
RESEARCH TRIANGLE PARK, NC 27711

DISCLAIMER

This report has been reviewed by the Environmental Sciences Research Laboratory, U.S. Environmental Protection Agency, and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the U.S. Environmental Protection Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.

AFFILIATION

Dr. Shreffler, the Project Officer, is on assignment to the Meteorology and Assessment Division, Environmental Sciences Research Laboratory, from the National Oceanic and Atmospheric Administration, U.S. Department of Commerce.

ABSTRACT

A set of instructions have been compiled for use of a Lagrangian photochemical air quality simulation model adapted to the St. Louis, Missouri/Illinois metropolitan region and the Regional Air Pollution Study (RAPS) data base. The computer model, developed by Environmental Research and Technology, Inc., consists of a set of computer programs for the simulation of atmospheric transport, turbulent diffusion, and chemical kenetics of photochemical pollutants. The model is used to predict atmospheric concentrations of ozone, nitrogen dioxide, carbon monoxide, sulfur dioxide, and sulfate within an air column moving at the mean wind speed.

Descriptions of the meteorological, source emissions, and air quality data requirements, as well as sample input and output files, are provided. The computational procedures for using the model and a listing of the computer code are included.

CONTENTS

ABSTRACT iii

LIST OF ILLUSTRATIONS viii

LIST OF TABLES x

1. INTRODUCTION 1-1

2. COMMON FEATURES OF THE CODE MODULES 2-1
   - 2.1 Reading the Data Cards 2-1
   - 2.2 Systems of Units and Time-Date Conventions 2-1
   - 2.3 Specification of Locations 2-4

3. THE METEOROLOGICAL MODULE 3-1
   - 3.1 Raw Data Requirements 3-1
   - 3.2 Control Parameters and Input Data 3-2
   - 3.3 Specification of the Station Measurement Data 3-17
   - 3.4 Modeling Region and Emissions Grid 3-20
   - 3.5 Selection of the Vertical Mesh Point Elevations 3-24
   - 3.6 Description of the Output 3-24
   - 3.7 Internally Specified Data 3-39
   - 3.8 Surface Roughness Length 3-39
   - 3.9 Printer-plot Boundaries 3-43
   - 3.10 Computational Procedures 3-44
     - 3.10.1 Calculation of Air Parcel Trajectories 3-44
     - 3.10.2 Interpolation of Other Atmospheric Parameters Along the Trajectory 3-48
     - 3.10.3 Calculation of Eddy Diffusivity Coefficients 3-48
   - 3.11 Program Modifications 3-49
     - 3.11.1 Changing the Number of Measurement Stations 3-49
     - 3.11.2 Changing the Azimuth Exclusion Sector 3-50

4. THE EMISSIONS MODULE 4-1
   - 4.1 Control Parameters and the Input Deck 4-1
     - 4.1.1 General Inputs 4-11
     - 4.1.2 Area Source Emissions Control Inputs 4-12
     - 4.1.3 Point Source Emissions Control Inputs 4-12
   - 4.2 Description of the RAPS Input Files 4-13
     - 4.2.1 Emissions Grid File 4-13

Preceding page blank

CONTENTS (CONTINUED)

4.2.2 Area Source Emissions File 4-14
4.2.3 Point Source Emissions File 4-15
4.3 Description of the Outputs 4-16
4.3.1 Area Source Emissions Outputs 4-16
4.3.2 Point Source Emissions Outputs 4-20
4.3.3 Punched Outputs 4-27
4.4 Computational Procedures 4-27
4.4.1 Area Source Emissions Tabulations 4-27
4.4.2 Point Source Emissions Tabulations 4-29
4.5 Internally Specified Data 4-36
4.6 Program Modifications 4-36
5. THE CHEMICAL-DIFFUSION MODULE 5-1
5.1 Control Parameters and Input Data 5-1
5.1.1 Source Emissions Data 5-25
5.1.2 Control Parameters and the Vertical Mesh 5-26
5.1.3 Chemical Species List 5-28
5.1.4 Emission Species and Surface Deposition Species 5-28
5.1.5 Initial Concentrations 5-29
5.1.6 Chemical Reaction Mechanism Inputs 5-29
5.1.7 Photodissociation Rate Input Parameters 5-31
5.1.8 Sky Clearness, Temperature, and Diffusivity Schedules 5-32
5.2 ERT Photochemical Mechanism 5-33
5.3 Photodissociation Rates 5-33
5.4 Description of the Outputs 5-39
5.5 Computational Procedures 5-54
5.6 Program Modifications 5-56
5.6.1 Changing the Chemical Mechanism 5-56
5.6.2 Changing the Partitioning of Emissions Input Data 5-59
5.6.3 Increasing the Number of Species with Emission Inputs 5-60
5.6.4 Changing the Plot Species 5-60

CONTENTS (CONTINUED)

6. REFERENCES 6-1

APPENDIX A CODE COMMON BLOCK LOCATIONS
  1. Meteorological Module Common Blocks
  2. Emissions Module Common Blocks
  3. Chemical-Diffusion Module Common Blocks

APPENDIX B CODE SUBROUTINE DESCRIPTIONS
  1. Meteorological Module Subroutines and Their Use
  2. Emissions Module Subroutines and Their Use
  3. Chemical-Diffusion Module Subroutines and Their Use
  4. Utility Library Subroutines and Their Use

APPENDIX C CODE FORTRAN SOURCE LISTINGS
  1. Meteorological Module Listings
  2. Emissions Module Listings
  3. Chemical-Diffusion Module Listing
  4. Utility Library Listing

APPENDIX D AN IMRPOVED PHOTOCHEMICAL MECHANISM AND RELATED MODIFICATIONS TO THE CHEMICAL-DIFFUSION MODULE
  1. Description of Chemical Mechanism and Related Modifications
  2. KEMOD Input Data Modifications
  3. KEMOD FORTRAN Source Code Modifications

LIST OF ILLUSTRATIONS

Figure		Page
1-1	Physical Concept of the Trajectory Model	1-2
1-2	Data Processing Flow Chart	1-4
2-1	Process for Reading Data Cards	2-5
3-1	Meteorological Module Sample Problem Input Listing	3-8
3-2	METMOD Input Variable Sequence and Options	3-13
3-3	The IVG Array	3-23
3-4	Meteorological Module Sample Output	3-26
3-5	Meteorological Module Sample Output	3-27
3-6	Meteorological Module Sample Output	3-28
3-7	Meteorological Module Sample Output	3-29
3-8	Meteorological Module Sample Output	3-30
3-9	Meteorological Module Sample Output	3-31
3-10	Meteorological Module Sample Output	3-33
3-11	Meteorological Module Sample Output	3-34
3-12	Meteorological Module Sample Output	3-35
3-13	Meteorological Module Sample Output	3-36
3-14	Meteorological Module Sample Output	3-37
3-15	Meteorological Module Sample Output	3-38
3-16	Surface Roughness Length Profile	3-42
3-17	Reverse Trajectory Development	3-47
4-1	Emissions Module Sample Problem Input Listing	4-5
4-2	EMMOD Input Variable Sequence and Options	4-10
4-3	Emissions Module Sample Output	4-17
4-4	Emissions Module Sample Output	4-18
4-5	Emissions Module Sample Output	4-19

LIST OF ILLUSTRATIONS (CONTINUED)

Figure		Page
4-6	Emissions Module Sample Output	4-21
4-7	Emissions Module Sample Output	4-22
4-8	Emissions Module Sample Output	4-23
4-9	Emissions Module Sample Output	4-24
4-10	Emissions Module Sample Output	4-25
4-11	Emissions Module Sample Output	4-26
4-12	Emissions Module Sample Output	4-28
4-13	Plume Entrainment in a Lagrangian Air Parcel	4-31
4-14	Lateral Spreading of Plumes Into and Out of Moving Air Parcels	4-32
4-15	Vertical Distribution of Emission Below Inversion Base	4-33
5-1	Chemical-Diffusion Module Sample Problem Input Listing	5-7
5-2	KEMOD Input Variable Sequence and Options	5-24
5-3	Chemical-Diffusion Module Sample Output	5-42
5-4	Chemical-Diffusion Module Sample Output	5-43
5-5	Chemical-Diffusion Module Sample Output	5-44
5-6	Chemical-Diffusion Module Sample Output	5-46
5-7	Chemical-Diffusion Module Sample Output	5-47
5-8	Chemical-Diffusion Module Sample Output	5-48
5-9	Chemical-Diffusion Module Sample Output	5-49
5-10	Chemical-Diffusion Module Sample Output	5-50
5-11	Chemical-Diffusion Module Sample Output	5-52
5-12	Original and Revised Area Source Flux Schedules	5-53

LIST OF TABLES

Table		Page
2-1	Code Utility Subroutine Library	2-2
2-2	Modeling Units	2-2
2-3	Equivalent Clock Times	2-3
3-1	Meteorological Module Input Variable Description	3-3
3-2	The Digital Wind Azimuth System	3-18
3-3	The Station-measured Parameters in the St. Louis Input File	3-18
3-4	Variables Stored in the CON Array	3-21
3-5	Meteorological Module Internally Specified Data	3-40
4-1	Emissions Module Input Variable Description	4-2
4-2	Emissions Module Internally Specified Data	4-38
4-3	Plume Dispersion Coefficients For Non-Tall Stacks	4-40
4-4	Plume Dispersion Coefficients For Tall Stacks	4-41
5-1	Chemical-Diffusion Module Input Variable Description	5-2
5-2	User Specified and Internally Computed Initial Concentrations	5-30
5-3	The ERT Photochemical Reaction Mechanism	5-34
5-4	Chemical Species Symbol Designations	5-37
5-5	NO_2 Photodissociation Rates	5-40
5-6	HCHO Photodissociation Rates	5-41

1. INTRODUCTION

This manual contains instructions for using the Lagrangian trajectory model developed by Environmental Research and Technology, Inc. (ERT). The model consists of a set of computer programs for simulation of Lagrangian transport and atmospheric reactivity of photochemical pollutants. These instructions describe features of the code which a user must understand in order to successfully utilize the model. The manual describes the code as adapted to the St. Louis, Missouri region and the Regional Air Pollution Study (RAPS) data base. Suggestions are provided for adapting the model to regions with different data base structures.

The Lagrangian trajectory code is written in the FORTRAN IV computer language in the single precision mode. The code has 3 main programs, 82 subroutines, 11 functions and block data programs. The maximum storage requirement of any one program is 51,000 words for execution.

A diagram of the physical concept of the model is shown in Figure 1-1. An air parcel is advected along a path (trajectory) determined by the local prevailing wind speed and direction. The air parcel is divided into a mesh of variably spaced points in the vertical direction. As the air parcel moves along a trajectory, it receives primary pollutant emissions from surface and elevated sources. The pollutants within the air parcel undergo chemical reactions and vertical diffusion, with photolysis by ultraviolet radiation driving several of the chemical transformations that occur. At each mesh point in the parcel, concentrations of 30 chemical species are computed. These species include ozone, oxides of nitrogen and sulfur, hydrocarbons, carbon monoxide and radical intermediates of known importance in the formation of photochemical smog. The output of the model consists of these calculated concentrations as functions of time and height along the trajectory.

The model is designed with a modular structure corresponding to the three data processing tasks required to compute atmospheric pollutant concentrations. The first task consists of processing meteorological data to determine air trajectories and characterize atmospheric conditions along the trajectories. This task is performed by the meteorological module of the code, program METMOD. The second task involves processing area source and point source emissions data to assemble

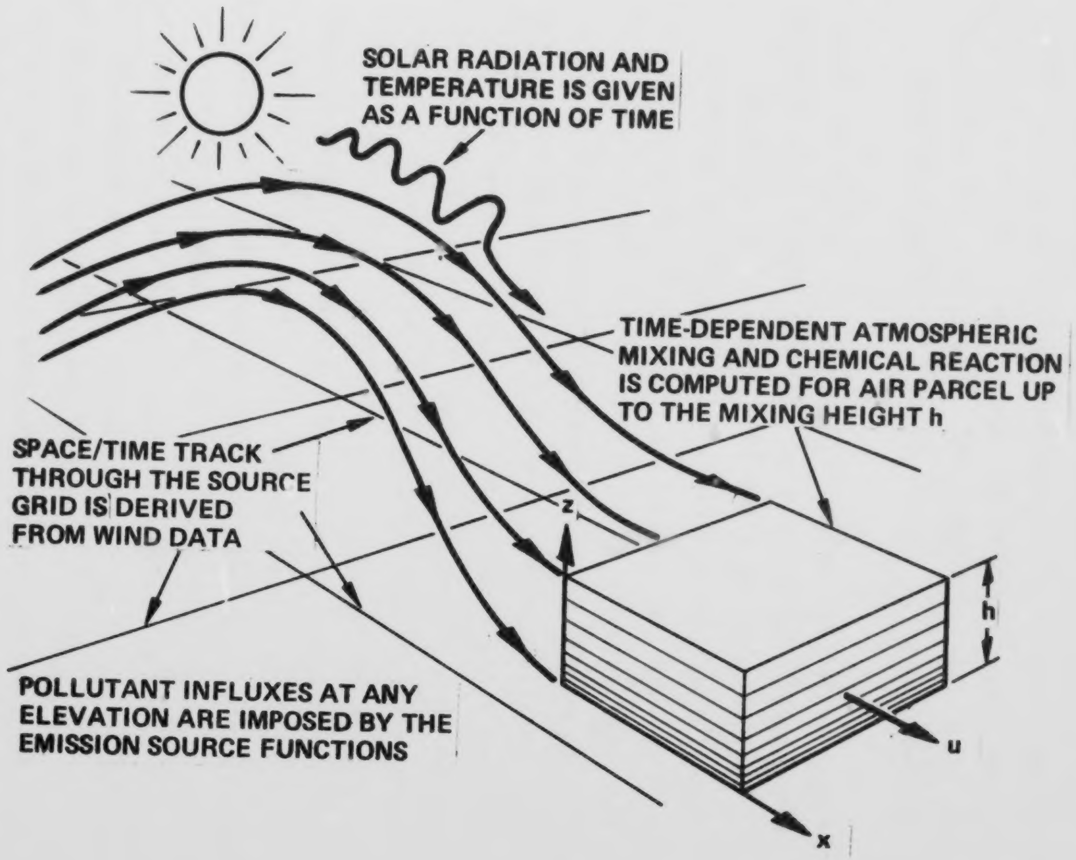

SOLAR RADIATION AND TEMPERATURE IS GIVEN AS A FUNCTION OF TIME
TIME-DEPENDENT ATMOSPHERIC MIXING AND CHEMICAL REACTION IS COMPUTED FOR AIR PARCEL UP TO THE MIXING HEIGHT h
SPACE/TIME TRACK THROUGH THE SOURCE GRID IS DERIVED FROM WIND DATA
z
h
POLLUTANT INFLUXES AT ANY ELEVATION ARE IMPOSED BY THE EMISSION SOURCE FUNCTIONS
u
x

Figure 1-1 Physical Concept of the Trajectory Model

emission rate schedules for the various pollutants along the trajectories. This task is performed by the emission module, program EMMOD. The last task is the computation of the solution to the diffusion equation, with simultaneous chemistry, for the species concentrations along the trajectory. This task is performed by the chemical-diffusion module of the code, program KEMOD.

The data processing activities proceed in the sequence of the tasks described above, as schematically illustrated in Figure 1-2. The user assembles wind, temperature, and ultraviolet radiation data for a single day in the modeling region. A start location, vertical mesh point elevations, and several program control parameters are selected. This data is used by the meteorological module to determine temporal schedules of wind speed and direction, atmospheric mixing coefficients (K_z) and stability, surface temperature, and sky clearness. The trajectory start location, wind speed and direction, surface temperature, and atmospheric stability outputs are transferred to the emissions module input stream. This data is used in addition to the gridded area source emission rates, point source locations, emission rates, and stack parameters, and a schedule of mixing heights by EMMOD to generate the emission source schedule for the trajectory. This emission schedule is then transferred to the chemical-diffusion module input stream along with the temperature, sky clearness, and mixing coefficients output from METMOD. The chemical-diffusion module's input stream is complete upon user specification of initial pollutant concentrations, chemical reaction rates, and several control parameters. The data processing activities are concluded by running the chemical-diffusion module, which outputs the computed concentration profiles.

It should be noted that the meteorological and emissions modules have been equipped with input data retrieval software, which is specific to the RAPS data tape format. Much of the input data is retrieved from the RAPS tapes, not the card-image input stream. Modification of this software is required for application of the model in regions other than St. Louis. The chemical diffusion module input stream is read entirely from the card reader and thus does not bear this restriction. However, it has several subroutines which are specific to the ERT photochemical mechanism.

ERT TRAJECTORY MODEL

DATA PROCESSING FLOW CHART

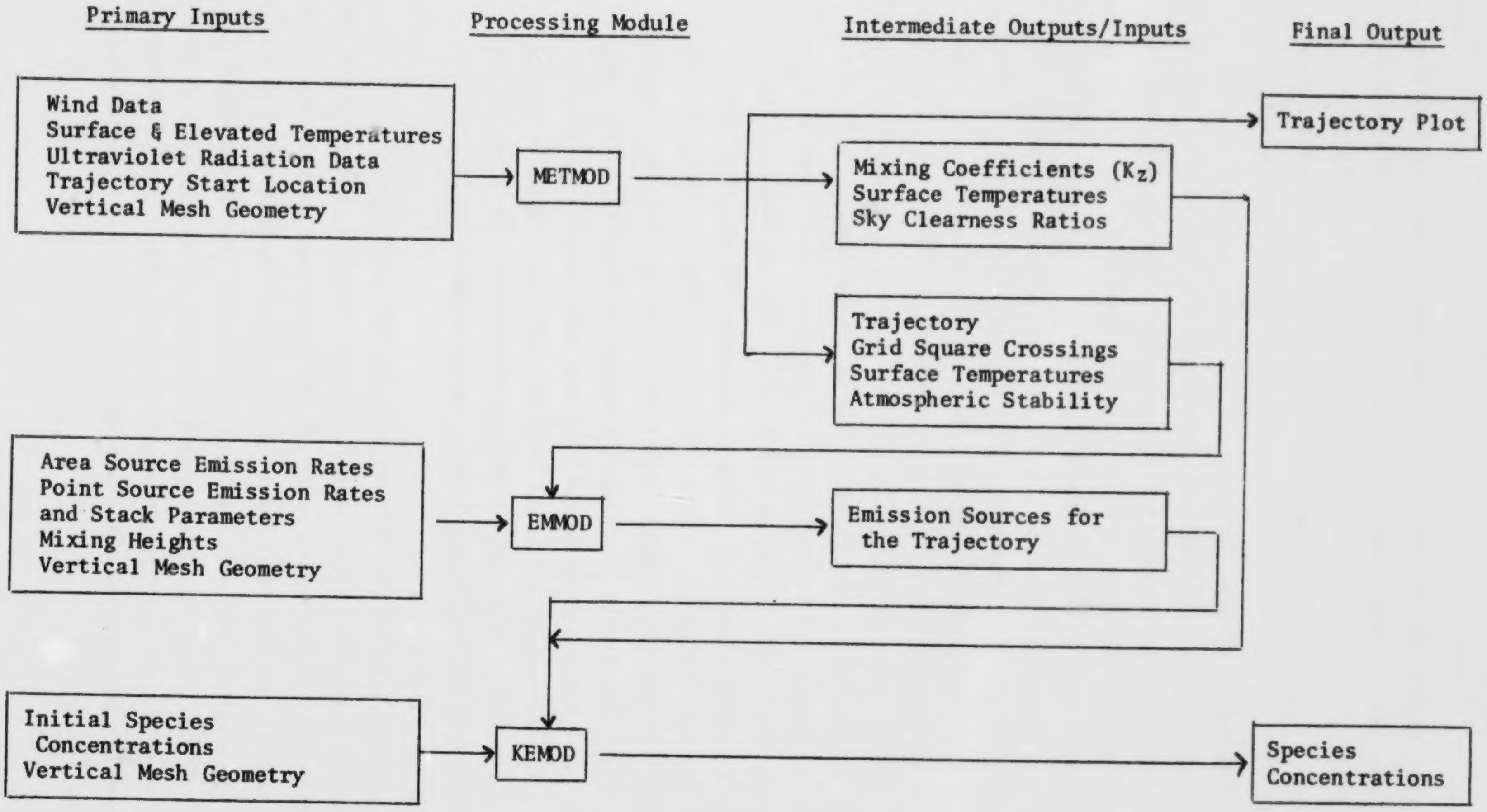


Figure 1-2

The sections of this manual that follow describe the operation of the three modules of the code. Appendix A lists the common block locations in the code. Appendix B includes a description of the programs, subroutines, and functions used by the code. Appendix C contains the FORTRAN source listing of the complete code as run on Control Data Corporation computers. Appendix D describes an alternate chemical mechanism and related modifications for its use.

2. COMMON FEATURES OF THE CODE MODULES

The three modules of the code have certain common features and use subroutines from a common utility program library. This section describes the common card reading process, system of units, time-date conventions, and coordinate systems used by the code. In addition, a list of utility library subroutines is provided in Table 2-1.

2.1 Reading the Data Cards

The process of reading the input cards is illustrated in Figure 2-1. The data cards are read by subroutine PREDAT from the card reader, which is referred to as logical unit 5 [e.g., READ (5,N) LIST]. PREDAT reads the data until it detects an end-of-file mark or a card with the word "MORE" in columns 1-4. It then produces a listing of the data deck on the line printer, which is designated as logical unit 6 [e.g., WRITE (6,N) LIST]. Examples of the output produced by PREDAT are contained in Sections 3, 4 and 5. The subroutine also writes the data on a scratch peripheral disk file designated as logical unit number 3. After writing the data on unit 3, the subroutine rewinds unit 3, and the data are then ready to be read by the program. Each module of the code calls PREDAT at the beginning of the main program.

2.2 Systems of Units and Time-Date Conventions

The model performs its calculations in metric units. Table 2-2 lists the units used by the code. Clock-time is expressed on two different clocks. The model uses the 0-2400 hours military clock and a 0-1440 minutes from midnight clock. Table 2-3 lists some equivalent times on the two clocks. Dates are generally expressed on a year-month-day basis (e.g., YRMODA = 760613 is June 13, 1976). In situations where dates are specified on a Julian basis, leap year corrections are included.

TABLE 2-1

CODE UTILITY SUBROUTINE LIBRARY

Subroutine	Module Requirements		
	METMOD	EMMOD	KEMOD
FMINF	X	X	
MCHAR	X		X
MDATE	X	X	X
NEWPAG	X	X	X
PREDAT	X	X	X
SECOND	X	X	X
SETPLT	X		X
SOLAR	X		X
XMIT	X	X	X

TABLE 2-2

MODELING UNITS

Quantity	Units
Length - vertical	meters
Length - horizontal	kilometers
Time	minutes and hours
Mass	kilograms
Concentration	parts per million
Temperature	degrees Celsius
Luminous intensity	Langleys per minute

TABLE 2-3

EQUIVALENT CLOCK TIMES

Time	0-2400 Hour Clock	0-1440 Minutes Clock
6 AM	600.	360.
7 AM	700.	420.
8 AM	800.	480.
9 AM	900.	540.
10 AM	1000.	600.
11 AM	1100.	660.
12 PM	1200.	720.
1 PM	1300.	780.
2 PM	1400.	840.
3 PM	1500.	900.
4 PM	1600.	960.
5 PM	1700.	1020.
6 PM	1800.	1080.
6:59 PM	1859.	1139.

2.3 Specifications of Locations

The program uses both a global and local coordinate system. Since the convention in air pollution modeling is to use the Universal Transverse Mercator (UTM) coordinate system, the program uses the label UTM for all global coordinate system data. All user specified location coordinates are input in the global coordinate system. The user also specifies an origin for a local coordinate system used internally by the programs.

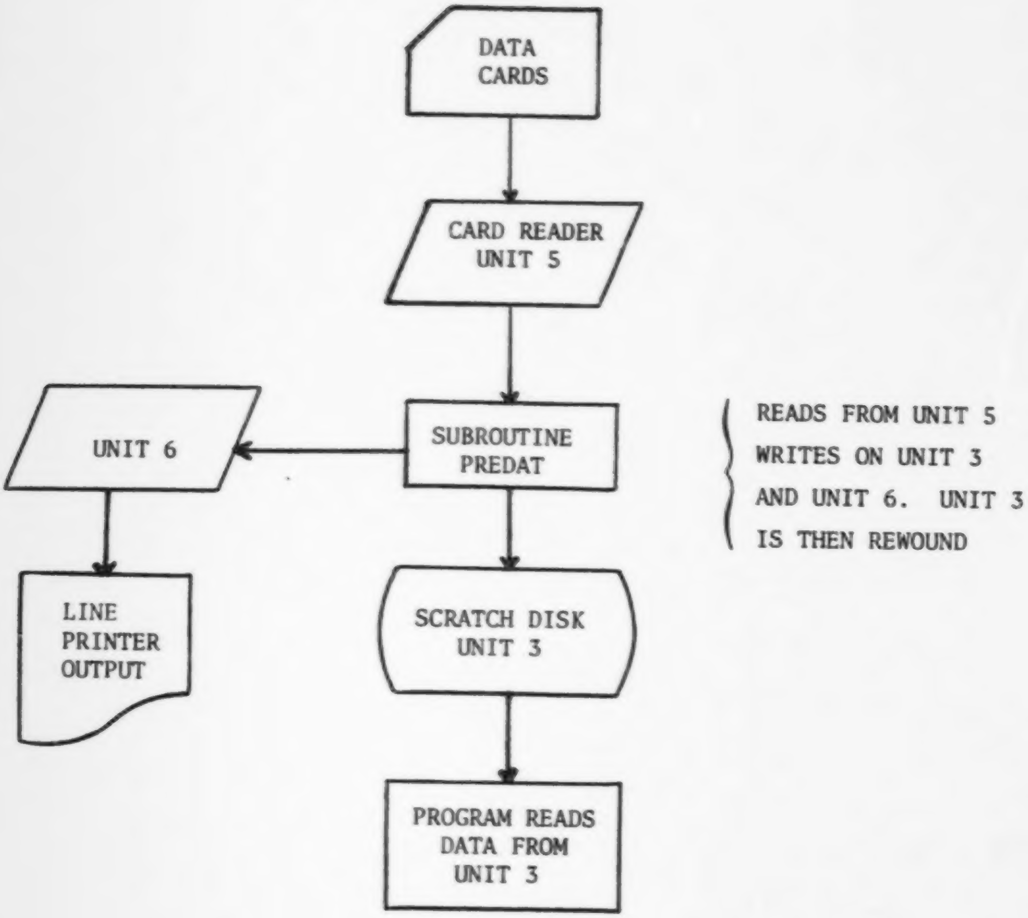


Figure 2-1 Process for Reading Data Cards

3. THE METEOROLOGICAL MODULE

This section describes procedures for using the program METMOD. METMOD generates air parcel trajectories and characterizes atmospheric conditions along the trajectories from basic meteorological data. Raw data requirements, input control parameters, input structures, optional modes of operation, output, and the basic computational procedures are described. Listings of the input and output for a sample problem are included. Since this program requires internal specification of data, which is characteristic of a particular modeling region, the user is encouraged to examine the FORTRAN source listing in Appendix C if he/she is adapting the model for applications to a region other than St. Louis.

3.1 Raw Data Requirements

The meteorological module requires a minimum input data base consisting of the following information:

- Hourly wind speeds and directions (from a network of monitoring stations)
- Hourly surface temperatures (regionally representative)
- Early morning and afternoon vertical temperature profiles (regionally representative).

The fidelity of the modeling is enhanced by using a data base with maximum spatial and temporal resolution. The program has the capability to utilize a data base, which includes the following data from a network of aerometric measurement stations:

- Hourly surface winds
- Hourly surface temperatures
- Vertical temperature profiles (three per day)
- Ultraviolet radiation data
- Ground-level pollutant concentrations

The optional ultraviolet radiation data are important to the modeling when the user intends to simulate days with unclear sky conditions. The pollutant concentrations, on the other hand, are input to the program more as a convenience to the user. They are interpolated along the trajectory in a manner similar to the wind data to approximate the measured concentrations in the Lagrangian reference frame. This computation is, of course, optional but quite useful when the user selects initial pollutant concentrations for the chemical-diffusion module.

3.2 Control Parameters and Input Data

The input parameters of the program will be discussed in the general order in which they are read by the code. The reader should refer to Figure 3-1, which contains a copy of the listing produced by PREDAT, to obtain a better picture of the actual structure of the data deck. Table 3-1 shows a list of the program inputs, along with a description of the function of each input variable, the name of the variable used in the code, and the format. In addition, Figure 3-2 contains a flow chart illustrating the sequence in which the variables are read under the different modes of operation.

It is important for the user to recognize that the selection of trajectories is often an iterative process. The user will usually desire the trajectories to start or end at a particular station or subregion of the modeling region and to have as long a duration as possible. Satisfying these requirements usually requires generation of several trajectories for different start times and/or start locations. The METMOD program is designed with a multiple-case feature, which allows generation of multiple trajectories in a single computer run.

Several modes of operation of the code are available to the user. The variables, which control the selection of these operational modes, are read immediately following the output page label (TITLE) in the input deck. When the user wishes the trajectory to be generated from wind data, he/she sets the input parameter TRAJEX equal to NO. This is the code's most common mode of operation. Alternatively, the user has the option to input a trajectory which has been externally computed. In this case, the user sets TRAJEX = YES and inputs a trajectory start location (input 23) and a schedule of times, wind speeds, and directions (input 24). In either of these cases, the user can request or suppress

TABLE 3-1

METEOROLOGICAL MODULE INPUT VARIABLE DESCRIPTION

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
1	TITLE	Label for each page of output.	(20A4)	May contain any user comments in columns 1-80.
2	TRAJEX	Wind trajectory is input externally (YES) or computed internally (NO).	(40X,A4)	This is a Hollerith variable which equals YES or NO. In either case the word must start in column 41.
3	KPWIND	Output for other code modules punched (YES) or not (NO).	(40X,A4)	See comments for input number 2.
4	NEEDAQ	Interpolation of extra atmospheric parameters is desired (YES) or not (NO).	(40X,A4)	See comments for input number 2.
5	NEEDKZ	Calculation of eddy diffusivity coefficients is desired (YES) or not (NO).	(40X,A4)	See comments for input number 2 and 26-33.
6	LTAPE	Logical unit number of meteorological data file.	(40X,I2)	Equals 3 for cards.
7	KPSTAT	Output a list of the measurement station names and coordinates (YES or NO).	(40X,A4)	See input number 2. Required first case only.
8	KXTRA	Print the extra trajectory output data (YES) or not (NO).	(40X,A4)	See input number 2. Required first case only. (See Section 3.7)
9	CONVRT	Multiplier needed to convert the wind speeds to km/hr.	(40X,F10.0)	Required first case only.

TABLE 3-1 (Continued)

Input Number	Variable Names Used in Code	Variable Function	FORTRAN Format	Remarks
10	UTMXOR, UTMYOR	Global x and y coordinates of the emissions grid origin (km).	(40X,2F10.0)	Required first case only.
11	DELX, DELY	Minimum grid square size (km) in the x and y directions.	(40X,2F10.0)	Required first case only.
12	NX, NY	Number of grid squares in the x and y directions.	(40X,2I10)	Required first case only.
13	RMIN, RMAX	Minimum and Maximum Distances used in the Interpolation Scheme (km).	(40X,2F10.0)	Required first case only.
14	IDT, XLCL	Data and time at the start of the trajectory	(40X,I6,4X, F4.0)	Date on the YRMODA basis and time on the 0-2400 clock. Only the date is read if TRAJEX = YES.
15	STIME	Is the start time STANDARD and DAYLIGHT time?	(40X,A4)	This is a Hollerith variable equal to STAN or DAYL.
16	ID, XS, YS	Trajectory start point location which can either be a station name or the global x and y coordinates.	(40X,2A4,2X, 2F10.0)	Do not specify both a station name and coordinates.
17	TTOTAL	Desired trajectory duration (hours).	(40X,F10.0)	
18	DTSEG	Time interval between trajectory wind calculations or nodes (hours).		Must be an even multiple of the measurement data interval.
19	AZ, VEL	Azimuth (degrees CCW from east to vector parallel to wind) and velocity (km/hr) of initial wind vector.	(40X,2F10.0)	These optional inputs override the initial wind vector calculation. Blanks or zeros yields normal operations.

TABLE 3-1 (continued)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
20	ITF	Trajectory is to be generated forward (+1) or backward (-1) in time.	(40X,I2)	
21	IRF	Station measurement weighted as 1/R if (01) or $1/R^2$ if (02).	(40X,I2)	Required with TRAJEX YES or NO.
22	NC	Number of nearby measurement stations desired for interpolation of station data.	(40X,I2)	$1 \leq NC \leq 3$ required with TRAJEX = YES or NO.
23	P(1,1), P(2,1)	Global x and y coordinates of the trajectory start point for externally input trajectory (km).	(40X,2F10.0)	Required only with TRAJEX = YES. See Figure 3-2 for its input position.
24	T, V, TH	Arrays of the externally input trajectory times, wind velocity, and wind azimuth (degrees CCW from east parallel the wind vector).	(40X,3F10.0)	Required only with TRAJEX = YES. See Figure 3-2 for these input positions. Time is on the 0-2400 basis. The last card should be a negative entry in columns 41-50. A maximum of 100 cards may be input.
25	XR, XL, YB, YT	Left, right, bottom, and top boundaries of the printer-plot (km).	(30X,4F10.0)	In local coordinates as described in Section 3.10.
26	NKZDAT	Read a new set of input data for the eddy diffusivity calculation (YES) or (NO).	(40X,A4)	See input number 2. Must equal YES in first data set with K_z calculation desired.

TABLE 3-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
27	KZPRTX	Print detailed K_z vertical profiles for each hour as an extra output (YES) or not (NO).	(40X,A4)	See input number 2 and Section 3.7.
28	TIMESD	Time of the vertical temperature profile.	(40X,F10.0)	One card with time on the 0-2400 clock precedes each vertical temperature profile input.
29	ZELVSD, TEMPSD	Arrays of elevations (meters) and corresponding temperatures (degrees C).	(40X,2F10.0)	Each card contains one elevation and temperature. Each profile may contain up to 50 cards. Up to 3 profiles are input, with a negative entry in columns 41-50 after each profile, and an extra negative entry after the last profile.
30	RDTEMP	An externally specified schedule of surface temperature is to be read (YES) or not (NO).	(40X,A4)	See input number 2. If RDTEMP = NO, the temperatures must be previously interpolated from station data.
31	A, TEMPSF	A schedule of times and externally specified surface temperatures (degrees C).	(40X,2F10.0)	One time on the 0-2400 clock and temperature per card. The data must be specified for every hour following the first profile release time through the last trajectory node time. Last card has a negative entry in columns 41-50.

TABLE 3-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
32	NOHTS	Number of vertical mesh points.	(40X,I10)	The maximum and recommended number of mesh points is 5. See Section 3.5.
33	CELHTS	Elevations above ground of vertical mesh points (meters).	(40X,F10.0)	One elevation per card, input by ascending elevations. See Section 3.5.
34	TERM	Last card for a single data set which equals MORE or END.	(A4)	TERM should equal MORE for all but the last data set in multiple case runs.

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
   1 ERT METEORELOGICAL MODULE     --      ST. LOUIS      --    6/29/76
   2 IS TRAJECTORY INPUT EXTERNALLY          NO
   3 DO WE WANT THE OUTPUTS   PUNCHED        YES
   4 DO WE WANT THE AIR QUALITY DATA         YES
   5 DO WE WANT DIFFUSIVITIES                YES
   6 LOGICAL UNIT OF MET-DATA PERIPHERRAL    21
   7 PRINT STATION COORDS                    YES
   8 DO WE WANT EXTRA TRAJECTORY OUTPUT      NO
   9 WIND SPEED CONVERSION FACTOR            3.6          (M/S TO KM/HR)
  10 UTM COORDINATES OF GRID ORIGIN    (KM)  680.00  4230.00

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  11 UNIFORM GRID SIZE (DX, DY IN KM)        1.0       1.0
  12 NUMBER OF UNIFORM GRID SQUARES (NX, NY)        100       100
  13 MINIMUM AND MAXIMUM INTERP RADIUS (KM)  1.0       75.0
  14 DATE AND TIME (YRMODA, 2400 HR CLOCK)   760629     600
  15 IS LOCAL TIME STANDARD OR DAYLIGHT      STANDARD
  16 START LOCATION (STATION ID OR UTM X,Y)            680.444   4224.050
  17 TRAJECTORY DURATION  (HOURS)            11.0
  18 TRAJECTORY SEGMENT LENGTH  (HOURS)      1.0
  19 STARTING AZIMUTH AND VELOCITY (OPTIONAL)
  20 DIRECTION FLAG (POS=FRWRD, NEG=BKWRD)    1

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  21 DISTANCE WEIGHTING FACTOR (1/R**  )     02
  22 NUMBER OF CLOSE STATIONS TO USE         03           (MAX = 3)
  23 PRINT - PLOT BOUNDARIES (KM)    -15.0    158.4      -16.0     115.0
  24 DO WE NEED VERT. TEMP PROFILES          YES
  25 DO WE WANT EXTRA KZ OUTPUT              YES
  26  RELEASE  TIME  (CST)          76-06-29   348.                UAN 142
  27  VERTICAL TEMPERATURE PROFILE  76-06-29   179.0     18.9      UAN 142
  28                                76-06-29   200.0     19.1      UAN 142
  29                                76-06-29   215.6     19.1      UAN 142
  30                                76-06-29   250.7     20.2      UAN 142

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  31                                76-06-29   295.1     23.3      UAN 142
  32                                76-06-29   300.0     23.4      UAN 142
  33                                76-06-29   313.0     23.8      UAN 142
  34                                76-06-29   400.0     23.2      UAN 142
  35                                76-06-29   448.5     22.9      UAN 142
  36                                76-06-29   485.0     23.8      UAN 142
  37                                76-06-29   500.0     23.8      UAN 142
  38                                76-06-29   549.0     23.8      UAN 142
  39                                76-06-29   600.0     23.6      UAN 142
  40                                76-06-29   670.1     23.3      UAN 142
```

Figure 3-1 Meteorological Module Sample Problem Input Listing

```
      1         11        21        31        41        51        61        71
 CARD +.........+.........+.........+.........+.........+.........+.........+.........
   41                                76-06-29    700.0     23.0       UAN 142
   42                                76-06-29    800.0     22.0       UAN 142
   43                                76-06-29    801.5     22.0       UAN 142
   44                                76-06-29    877.3     21.5       UAN 142
   45                                76-06-29    900.0     21.4       UAN 142
   46                                76-06-29    982.4     21.1       UAN 142
   47                                76-06-29   1000.0     20.9       UAN 142
   48                                76-06-29   1078.8     20.0       UAN 142
   49                                76-06-29   1100.0     19.8       UAN 142
   50                                76-06-29   1200.0     18.9       UAN 142

      1         11        21        31        41        51        61        71
 CARD +.........+.........+.........+.........+.........+.........+.........+.........
   51                                76-06-29   1300.0     18.0       UAN 142
   52                                76-06-29   1362.6     17.5       UAN 142
   53                                76-06-29   1400.0     17.4       UAN 142
   54                                76-06-29   1500.0     17.3       UAN 142
   55                                76-06-29   1522.3     17.3       UAN 142
   56                                76-06-29   1582.8     16.9       UAN 142
   57                                76-06-29   1600.0     16.7       UAN 142
   58                                76-06-29   1700.0     16.0       UAN 142
   59                                76-06-29   1766.1     15.4       UAN 142
   60                                76-06-29   1800.0     15.2       UAN 142

      1         11        21        31        41        51        61        71
 CARD +.........+.........+.........+.........+.........+.........+.........+.........
   61                                76-06-29   1900.0     14.4       UAN 142
   62                                76-06-29   1973.7     13.9       UAN 142
   63                                76-06-29   2000.0     13.7       UAN 142
   64                                76-06-29   2078.9     13.1       UAN 142
   65                                76-06-29   2100.0     12.9       UAN 142
   66                                76-06-29   2200.0     12.1       UAN 142
   67                                76-06-29   2300.0     11.3       UAN 142
   68                                76-06-29   2400.0     10.5       UAN 142
   69                                76-06-29   2423.0     10.3       UAN 142
   70                                76-06-29   2500.0      9.6       UAN 142

      1         11        21        31        41        51        61        71
 CARD +.........+.........+.........+.........+.........+.........+.........+.........
   71                                76-06-29   2600.0      8.7       UAN 142
   72                                76-06-29   2632.6      8.5       UAN 142
   73                                76-06-29   2688.4      8.1       UAN 142
   74                                76-06-29   2700.0      8.0       UAN 142
   75                                76-06-29   2800.0      7.3       UAN 142
   76                                76-06-29   2846.2      7.0       UAN 142
   77                                76-06-29   2900.0      6.6       UAN 142
   78                                76-06-29   3000.0      5.8       UAN 142
   79                                76-06-29   3064.2      5.3       UAN 142
   80                                76-06-29   3100.0      5.1       UAN 142
```

Figure 3-1 (Continued)

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  81                               76-06-29  3145.6       4.8      UAN 142
  82                                       -  10.0
  83  RELEASE  TIME  (CST)         76-06-29    945.                UAN 141
  84  VERTICAL TEMPERATURE PROFILE 76-06-29   149.0      27.6      UAN 141
  85                               76-06-29   161.1      26.7      UAN 141
  86                               76-06-29   200.0      26.5      UAN 141
  87                               76-06-29   288.7      26.0      UAN 141
  88                               76-06-29   300.0      25.8      UAN 141
  89                               76-06-29   351.9      25.3      UAN 141
  90                               76-06-29   379.1      23.9      UAN 141
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  91                               76-06-29   400.0      23.6      UAN 141
  92                               76-06-29   451.6      23.0      UAN 141
  93                               76-06-29   500.0      22.5      UAN 141
  94                               76-06-29   600.0      21.6      UAN 141
  95                               76-06-29   616.1      21.4      UAN 141
  96                               76-06-29   700.0      21.5      UAN 141
  97                               76-06-29   783.3      21.7      UAN 141
  98                               76-06-29   800.0      21.7      UAN 141
  99                               76-06-29   900.0      21.7      UAN 141
 100                               76-06-29  1000.0      21.7      UAN 141
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 101                               76-06-29  1001.6      21.7      UAN 141
 102                               76-06-29  1100.0      20.9      UAN 141
 103                               76-06-29  1195.5      20.1      UAN 141
 104                               76-06-29  1200.0      20.2      UAN 141
 105                               76-06-29  1283.9      20.4      UAN 141
 106                               76-06-29  1300.0      20.3      UAN 141
 107                               76-06-29  1400.0      19.8      UAN 141
 108                               76-06-29  1453.1      19.5      UAN 141
 109                               76-06-29  1500.0      19.1      UAN 141
 110                               76-06-29  1523.7      18.9      UAN 141
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 111                               76-06-29  1600.0      18.2      UAN 141
 112                               76-06-29  1700.0      17.3      UAN 141
 113                               76-06-29  1799.7      16.4      UAN 141
 114                               76-06-29  1800.0      16.4      UAN 141
 115                               76-06-29  1900.0      15.8      UAN 141
 116                               76-06-29  2000.0      15.2      UAN 141
 117                               76-06-29  2100.0      14.6      UAN 141
 118                               76-06-29  2125.7      14.5      UAN 141
 119                               76-06-29  2200.0      13.7      UAN 141
 120                               76-06-29  2300.0      12.7      UAN 141
```

Figure 3-1 (Continued)

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 121
 122                                76-06-29  2395.8      11.8        UAN 141
 123                                76-06-29  2400.0      11.8        UAN 141
 124                                76-06-29  2500.0      11.4        UAN 141
 125                                76-06-29  2527.7      11.2        UAN 141
 126                                76-06-29  2600.0      10.7        UAN 141
 127                                76-06-29  2700.0       9.9        UAN 141
 128                                76-06-29  2800.0       9.1        UAN 141
 129                                76-06-29  2876.3       8.5        UAN 141
 130                                76-06-29  2900.0       8.2        UAN 141
                                    76-06-29  2968.1       7.3        UAN 141
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 131
 132                                76-06-29  3000.0       7.1        UAN 141
 133                                76-06-29  3100.0       6.7        UAN 141
 134                                76-06-29  3153.9       6.4        UAN 141
 135                                         -  10.0
 136 RELEASE  TIME  (CST)           76-06-29  1553.                   UAN 141
 137 VERTICAL TEMPERATURE PROFILE   76-06-29   149.0      29.5        UAN 141
 138                                76-06-29   200.0      29.8        UAN 141
 139                                76-06-29   278.1      30.3        UAN 141
 140                                76-06-29   300.0      29.8        UAN 141
                                    76-06-29   379.1      27.8        UAN 141
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 141
 142                                76-06-29   400.0      27.4        UAN 141
 143                                76-06-29   480.4      25.7        UAN 141
 144                                76-06-29   500.0      25.6        UAN 141
 145                                76-06-29   554.5      25.2        UAN 141
 146                                76-06-29   600.0      24.5        UAN 141
 147                                76-06-29   610.3      24.3        UAN 141
 148                                76-06-29   700.0      23.4        UAN 141
 149                                76-06-29   703.9      23.4        UAN 141
 150                                76-06-29   800.0      22.2        UAN 141
                                    76-06-29   807.6      22.1        UAN 141
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 151
 152                                76-06-29   854.9      21.4        UAN 141
 153                                76-06-29   900.0      21.2        UAN 141
 154                                76-06-29   912.1      21.2        UAN 141
 155                                76-06-29  1000.0      20.7        UAN 141
 156                                76-06-29  1017.6      20.6        UAN 141
 157                                76-06-29  1065.9      20.1        UAN 141
 158                                76-06-29  1100.0      19.8        UAN 141
 159                                76-06-29  1200.0      19.0        UAN 141
 160                                76-06-29  1300.0      18.1        UAN 141
                                    76-06-29  1310.5      18.0        UAN 141
```

Figure 3-1 (Continued)

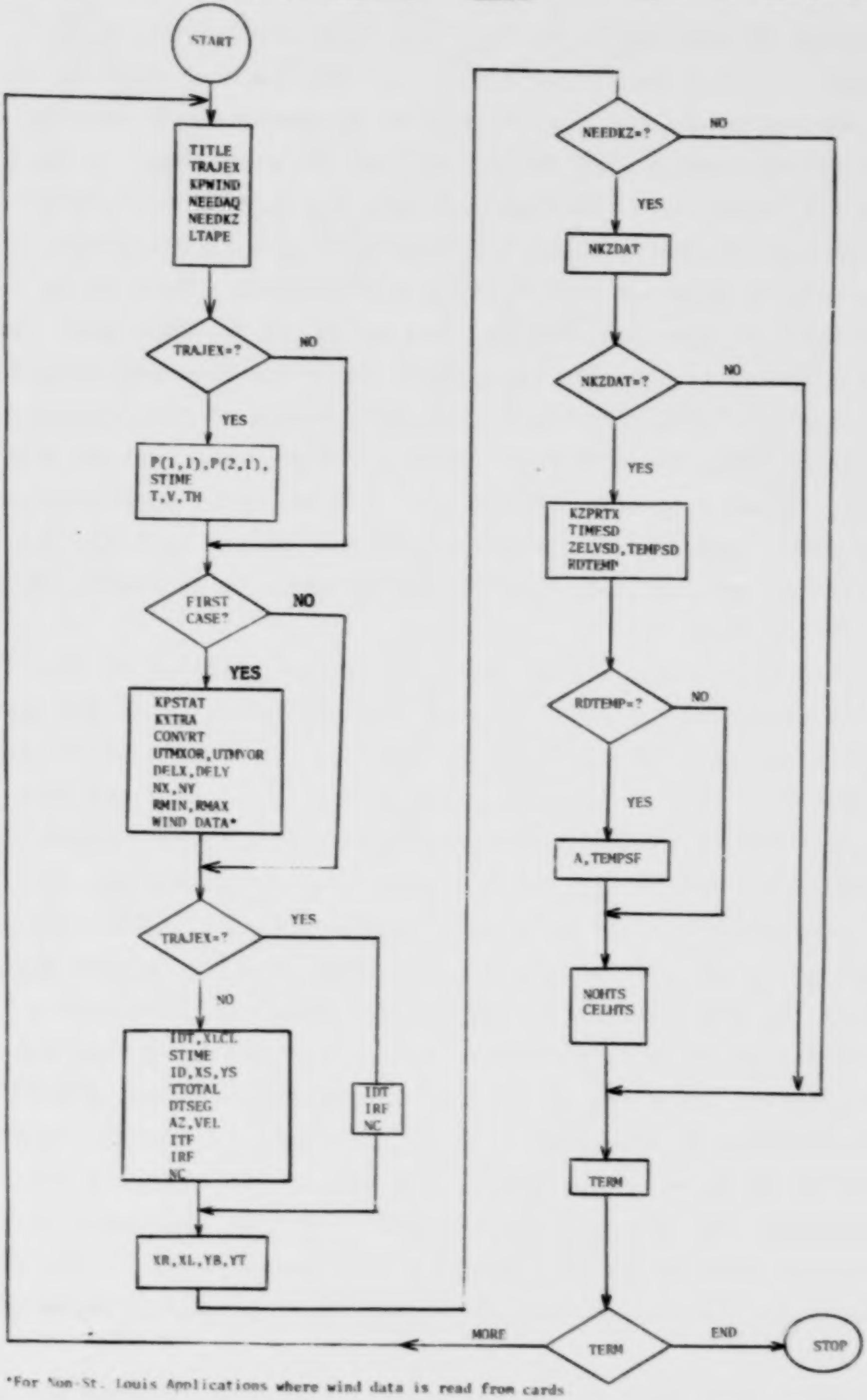
```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 161                               76-06-29  1400.0      17.4         UAN 141
 162                               76-06-29  1499.9      16.6         UAN 141
 163                               76-06-29  1500.0      16.6         UAN 141
 164                               76-06-29  1560.4      16.4         UAN 141
 165                               76-06-29  1600.0      16.2         UAN 141
 166                               76-06-29  1700.0      15.6         UAN 141
 167                               76-06-29  1800.0      14.9         UAN 141
 168                               76-06-29  1900.0      14.3         UAN 141
 169                               76-06-29  1972.9      13.9         UAN 141
 170                               76-06-29  2000.0      13.9         UAN 141

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 171                               76-06-29  2004.5      13.9         UAN 141
 172                               76-06-29  2100.0      13.2         UAN 141
 173                               76-06-29  2200.0      12.5         UAN 141
 174                               76-06-29  2300.0      11.8         UAN 141
 175                               76-06-29  2358.5      11.4         UAN 141
 176                               76-06-29  2400.0      11.2         UAN 141
 177                               76-06-29  2500.0      10.6         UAN 141
 178                               76-06-29  2567.7      10.1         UAN 141
 179                               76-06-29  2600.0      10.1         UAN 141
 180                               76-06-29  2700.0      10.0         UAN 141

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 181                               76-06-29  2713.5      10.0         UAN 141
 182                               76-06-29  2781.6       9.6         UAN 141
 183                               76-06-29  2800.0       9.5         UAN 141
 184                               76-06-29  2872.9       8.9         UAN 141
 185                               76-06-29  2895.9       9.2         UAN 141
 186                               76-06-29  2900.0       9.3         UAN 141
 187                               76-06-29  2930.5       9.4         UAN 141
 188                               76-06-29  3000.0       9.1         UAN 141
 189                               76-06-29  3000.2       9.1         UAN 141
 190                               76-06-29  3100.0       8.2         UAN 141

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 191                               76-06-29  3129.0       8.0         UAN 141
 192 LAST DUMMY NEGATIVE                     - 10.0
 193 SECOND LAST DUMMY NEGATIVE              - 10.0
 194 DO WE WANT EXTERNALLY INPUT SURF. TEMP. NO
 195 NUMBER OF KEMOD VERTICAL MESH POINTS    05
 196 MESH POINT ELEVATIONS (ABOVE SURFACE)   0.0         (METERS)
 197 2                                       120.0
 198 3                                       300.0
 199 4                                       600.0
 200 5                                       1200.0
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 201 END
```

Figure 3-1 (Continued)



*For Non-St. Louis Applications where wind data is read from cards

Figure 3-2 METMOD Input Variable Sequence and Options

the generation of eddy-diffusivity (K_z) coefficients by specifying NEEDKZ equal to YES or NO, respectively. If the user has supplied the surface temperature and air quality data by measurement station, the user sets NEEDAQ equal to YES for the program to interpolate the data along the trajectory. In addition, the user has the option to have the program produce the punched cards containing the METMOD outputs which are used as inputs to other modules of the code. This option is in effect when the user sets KPWIND equal to YES in the data deck. The use of this option is strongly recommended for error free data transfer.

The user next specifies the logical unit number of the peripheral device (LTAPE) where the wind data resides. If the wind data is read from cards, the user specifies LTAPE = 3. For St. Louis applications where the wind, temperature, radiation, and air quality data are retrieved from a tape, the user specifies LTAPE equal to the unit number assigned to the tape.

The program next reads a sequence of variables related to the code's initialization process. These variables are read only for the first case of a run with multiple case inputs. The first variable read in this series of cards is KPSTAT, which is set to YES when the user desires a printed listing of the measurement station names (stored internally). See Section 3.6 for a description of this output.

The program next reads an "extra" output parameter, KXTRA. When KXTRA equals YES, the output includes a listing of which station data are used in the interpolation of wind, temperature, and air quality data. A description of this extra output is included in Section 3.6.

The program next reads the wind speed conversion factor, CONVRT. The user must specify this factor, which is used to convert the user's wind data to the units of kilometers/hour used by the code. In the sample problem, this parameter has a value of 3.6 for conversion of the St. Louis wind data from meters/second to kilometers/hour. If the program reads a negative or zero value for CONVRT, it assumes a value of one.

The next three cards in the first-case-only series contain the grid origin coordinates (UTMXOR, UTMYOR), the minimum grid square sizes (DX, DY), and the number of minimum size grid squares (NX, NY) in the x and y directions. The reader is referred to Section 3.4 for an explanation of these parameters.

The next card read contains the minimum and maximum distances (RMIN and RMAX) used in the interpolation algorithm to screen data from stations very near to and very remote from a trajectory node. See Section 3.10 for a description of these parameters.

At this point the program completes the first-case-only initialization by either reading the wind data from cards or retrieving the winds, surface temperatures, radiation, and pollutant concentrations from tape. See Section 3.3 for a description of these inputs and the user's options.

The program next reads a series of cards with information concerning a specific trajectory. The first card in the series contains the date (IDT) and the start time (XLCL) of the trajectory. The date is specified using the year-month-day (YRMODA) convention. This card is followed by one indicating whether the start time is specified in daylight or standard time. Next, the program reads the start location. The user may specify either a station name (ID) or global x and y coordinates (XS, YS) to define this location. This card is followed by cards specifying the trajectory duration (TTOTAL) and the time interval between trajectory nodes (DTSEG) in hours. Next, the program reads a variable (ITF), which indicates whether the trajectory is to be generated in the forward or backward direction from the start location. This card is followed by one specifying whether the station measurements are to be weighted by reciprocal distances or by the squares of the reciprocal distances in the interpolation formula. The program next reads the number of close stations (up to 3) to be used in the interpolation of station data. The last card in this section specifies the printer-plot boundaries in local coordinates to be used in the trajectory plot. The reader is referred to Section 3.9 for instructions on proper selection of these boundary values.

When these cards have been read and the station measurements retrieved (from cards or tape), the program computes and plots the trajectory.

Following the trajectory generation and interpolation computations, the program calls the eddy diffusivity submodule (when NEEDKZ = YES). The submodule reads the remainder of the input data. The first card read here indicates whether or not the eddy diffusivity input cards should be read. This variable (NKZDAT) must equal YES in a first case data set and may equal NO in subsequent data sets where the same vertical temperature profiles and mesh point elevations are to be used. When the user specifies NKZDAT equal to YES, the program reads a flag for optional K_z output which determines whether or not the detailed (internal) eddy diffusivity profiles are to be printed. See Section 3.6 for a description of this output.

The eddy diffusivity submodule next reads up to three radiosonde release times and vertical temperature profiles. The input data is structured with one release time card followed by a set of cards with the elevations and temperatures from the sounding. These cards are ordered by ascending elevations until a negative elevation is read. One or two subsequent vertical temperature profiles may be similarly input in chronological order. A negative release time is specified after the final profile to terminate this operation. The user may specify the elevations as either height-above-ground or height-above-sea-level since the program subtracts the first elevation from all elevations prior to using the data.

Next, the program reads a variable (RDTEMP) that determines whether the hourly surface temperatures are to be read. This variable is set equal to NO when the user has supplied temperature data for the measurement stations, and the program has interpolated this data along the trajectory. When the network data has not been interpolated, or when the user wishes to override the interpolated temperatures, this flag is set to YES, and temperatures are read in as follows. The program reads a set of cards, each of which contains a time on the 0-2400 clock and the corresponding surface temperature. These cards are input in chronological order followed by a negative input for the time.

The last set of cards in the input deck specifies the number of vertical mesh points (NOHTS) and their elevations above ground (CELHTS). The program reads NOHTS elevation cards in order by ascending elevation. These vertical mesh parameters are used by all three modules of the code and require careful selection. Instructions are provided in Section 3.5 for their specification.

Finally, the program reads a variable (TERM) equal to MORE or END. When the user has stacked multiple data sets together for multiple trajectory generations, the user specifies MORE as the last card in each data set, except the last set where END is specified.

3.3 Specification of the Station Measurement Data

The manner in which the station measurement data is read by the code depends on the types of information available. In applications where the user has only wind data, the code reads the data from the card reader. In applications for locations such as St. Louis, where each station measures many useful parameters, the program reads the data from a tape. The data format differs significantly for the two cases.

When the wind data is read from the card reader, the code reads a pair of cards for each combination of station and date. The first card in the pair contains the date, station name, and the wind direction data. Columns 1-6 contain the date on the year-month-day basis. Columns 8-15 contain the station name. Columns 29-30 contain the number of compass sectors ($4 \leq n \leq 90$) utilized in the meteorological wind data reporting system under which the program interprets the azimuth data on the remainder of the card. Table 3-2 shows examples of the correspondence between these digitized wind azimuths and their polar angles. The twenty-four wind azimuths are specified in columns 33-80. The second card of the pair contains the digitized wind speed data in columns 33-80. The program assumes the wind data are hourly averages, with the first entry being for the 0000-0100 time period, the next for 0100-0200, etc. Blanks or negative values in the azimuth and speed fields are interpreted as being missing data. The program reads and stores the wind data for the stations for up to ten different dates. The reading of the data is terminated by reading one card with 999999 in columns 1-6.

In cases where a more extensive and disaggregated meteorological data base exists, different retrieval software is used to read, edit, and store the data. Since these types of data bases are rarely structured similarly, the user is encouraged to modify subroutine METIN to suit his/her application, using the St. Louis version as an example.

In the St. Louis application, the station data input file contains measured data for the parameters listed in Table 3-3. As indicated in the table, the program utilizes the wind, temperature, temperature gradient, solar radiation, and pollutant concentration measurements.

The input file is structured with a header record at the beginning of each hour which contains the year, julian day, and hour (0-23). The program converts the user-specified date in the input data deck to the

TABLE 3-2

THE DIGITAL WIND AZIMUTH SYSTEM

	North	East	South	West
Compass Heading of Wind Vector	0°	90°	180°	270°
Point No. (16 pt. system)	8	12	16	4
Point No. (90 pt. system)	45	67	90	22

TABLE 3-3

THE STATION-MEASURED PARAMETERS IN THE ST. LOUIS INPUT FILE

No.		Units	Utilized
1	Wind Speed	m/sec	YES
2	Wind Direction	degrees	YES
3	Temperature	degrees C	YES
4	Dew Point	degrees C	NO
5	Barometric Pressure	mb	NO
6	Temperature Gradient (30m-5m)	$\Delta T/25m$	YES
7	Solar Radiation, All(1)	Langleys/min	YES
8	Solar Radiation, $\lambda>3950A$	Langleys/min	YES
9	Solar Radiation, $\lambda>6950A$	Langleys/min	NO
10	Solar Radiation, All(2)	Langleys/min	NO
11	Ozone	ppm	YES
12	Carbon Monoxide	ppm	YES
13	Methane	ppm	YES
14	Total Hydrocarbons	ppmc	YES
15	Nitric Oxide	ppm	YES
16	Nitrogen Dioxide	ppm	YES
17	Nitrogen Oxides	ppm	YES
18	Total Sulfur	ppm	YES
19	Hydrogen Sulfide	ppm	NO
20	Sulfur Dioxide	ppm	YES
21	Light Scattering, BSCAT (1)		NO
22	Light Scattering, BSCAT (2)		NO

julian basis and proceeds if there is a match between it and the julian day read from tape. Each hour's data are structured with all the station values for the first parameters, followed by all the station values for the second parameters, etc., with the parameters ordered as shown in Table 3-2. The program reads individual station values for all 22 parameters for the first hour, then proceeds to edit and store the data it needs, then repeats this cycle until all the required data are stored. It is important to note that the program assumes that the station values are input in the order in which the station names are internally stored in subroutine SETIN.

The wind data are stored in the two-dimensional array AWDATA as described below.

AWDATA (1, Station Index) = Date (YRMODA)

AWDATA (2-3, Station Index) = Station Names

AWDATA (4, Station Index) = Number of wind sectors used in azimuth reporting system.

AWDATA (5-28, Station Index) = 24 hours of wind azimuths

AWDATA (29-52, Station Index) = 24 hours of wind speeds

These data are edited by placing a -1 in the array where missing data are indicated by very large values in the St. Louis input file. Also, the program makes a special adjustment to the St. Louis wind speed data from nine stations with short measurement towers. The program uses the regional average temperature gradient to infer atmospheric stability and adjusts the ten meter wind speeds to their thirty meter equivalents, using a stability dependent power-law vertical wind profile.

The program obtains the ultraviolet radiation by subtracting the solar radiation for wavelengths greater than 3950A from the total solar radiation data. In this application, the regional hourly average ultraviolet radiation is computed and stored in the array UV. The ultraviolet radiation data are not interpolated along the trajectory since only one-quarter of the stations measure this data.

The surface temperatures and pollutant concentrations are edited for missing data and stored in the array CON. This array is three-dimensional with its indices described as follows:

CON (I, J, K)

I = 1 thru 24 corresponding to hours 0 thru 23

J = station index corresponding to the order in which the stations are specified in subroutine SETIN and read in

K = parameter index

For the St. Louis application, the parameter index corresponds to one of the ten types of data listed in Table 3-4. The user specifies the number of parameters (NOSPEC) stored in CON and an array of the parameter names (CONAM) in data statements described in Section 3.7.

The user can store any data in this array which he/she wants interpolated along the trajectory. The most important data to store in the array are the surface temperature data, which should always be the first parameter (K=1).

3.4 Modeling Region and Emissions Grid

The user determines the geographic extent of the region being modeled. The region should be chosen large enough to allow for 8 to 12 hour trajectories for moderate wind speeds within its boundaries. It must extend far enough to include the major emission sources which affect air quality along the modeled trajectories.

Area source emissions data are compiled and allocated to a rectilinear grid, as described in Section 4. The meteorological module of the code determines a schedule of grid square crossing times and grid indices from the air parcel trajectories, so that the emission module can generate a schedule of emission fluxes along the trajectory. In order to do this, the program must be supplied with the x and y coordinates of the grid origin (UTMXOR, UTMYOR), the number (NX, NY) of fixed-size grid squares in the x and y directions, and the grid square size (DX, DY). In the sample problem, the origin is specified as follows:

UTMXOR = 680.(km)   UTMYOR = 4230.(km)

DX = 1.(km)   DY = 1.(km)

NX = 100   NY = 100

TABLE 3-4

VARIABLES STORED IN THE CON ARRAY

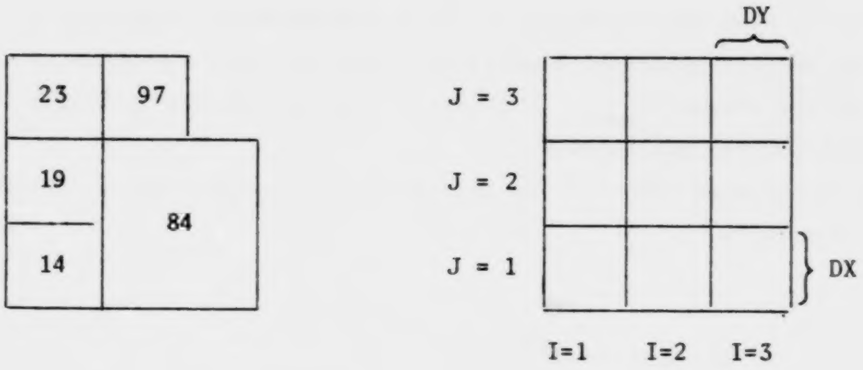
K Index	Parameters	Names
1	Temperature	(TEMP)
2	Ozone	(O3)
3	Carbon Monoxide	(CO)
4	Methane	(CH4)
5	Total Hydrocarbons	(THC)
6	Nitric Oxide	(NO)
7	Nitric Dioxide	(NO2)
8	Nitrogen Oxides	(NOX)
9	Total Sulfur	(TS)
10	Sulfur Dioxide	(SO2)

Thus, the area source emissions grid is a 100 x 100 kilometer region with its southwest corner located at UTM coordinates 680., 4230.

The area source emissions grid usually extends over the entire modeling region, but the program allows it to be a subregion within the modeling region. In other words, if the geographic distribution of significant area source emissions is of limited extent, the user is not constrained to specify an unnecessarily large grid. The user can internally specify (in subroutine EDGE) the width of a buffer zone which is added to all sides of the emission grid. Thus, the modeling region is defined by the area within the boundaries of the emissions grid plus the buffer zone. The program allows trajectories, monitoring stations, and emission point sources to be located anywhere within the modeling region. Nevertheless, the program uses the origin of the emissions grid as the origin of its local coordinate system.

The program has an additional feature that provides the capability of using area source emissions data allocated to variable size grid squares. In this situation, each grid square has a single identifier instead of the usual I and J indices on a fixed grid-square-size grid. This feature of the program is utilized in the St. Louis application where the area source emissions data has been allocated to grid squares ranging in size from 1 x 1 to 10 x 10 kilometers.

When this option is required, the user sets the variable ITGRID equal to YES in a data statement in the block data program. Then, the program calculates the schedule of grid square crossing times and single identifiers using its schedule of crossing times and (I, J) indices plus the IVG array. The IVG array contains the variable-size grid square identifiers for each (I, J) grid square in the emissions grid. As a preprocessing activity, the user must allocate the variable-size grid identifier to a grid with the minimum size squares. This information is then stored in a block data program for use in the code. Figure 3-3 shows an example of the correspondence between the variable-size grid square identifiers and the elements of the IVG array. Note that when there is no grid square identifier (i.e., no area source data) for a particular grid square, the corresponding element of the IVG array is set equal to zero.



IVG(I,J) = identifier of grid square I,J

IVG(1,3) = 23   IVG(2,3) = 97   IVG(3,3) = 0
IVG(1,2) = 19   IVG(2,2) = 84   IVG(3,2) = 84
IVG(1,1) = 14   IVG(2,1) = 84   IVG(3,1) = 84

Figure 3-3 The IVG Array

3.5 Selection of the Vertical Mesh Point Elevations

The vertical mesh point elevations (above ground) are selected on the basis of the meteorology of a particular day being modeled. The user must obtain or estimate the maximum mixing height along the trajectory. If this data is not explicitly reported by the meteorological monitoring network, the user should examine the afternoon vertical temperature profile(s) to find the elevation at which the potential temperature equals the surface potential temperature, and use this elevation as the maximum mixing height (Z_{max}). Typically, the maximum mixing height ranges from 500 to 2000 meters.

The recommended formulae for calcualting the mesh point elevations are provided below.

$$Z_1 = 0.0$$

$$Z_2 = \text{minimum of} \begin{cases} 60 \text{ meters} \\ .10* Z_{max} \end{cases}$$

$$Z_3 = \text{minimum of} \begin{cases} 200 \text{ meters} \\ .25* Z_{max} \end{cases}$$

$$Z_4 = \text{minimum of} \begin{cases} 700 \text{ meters} \\ .50* Z_{max} \end{cases}$$

$$Z_5 = \text{minimum of} \begin{cases} 2000 \text{ meters} \\ Z_{max} \end{cases}$$

Using these formulae will ensure maximum resolution in the lower portion of the air parcel where vertical concentration gradients are most important.

It should be point out that the vertical mesh point elevations used in the sample problem for the second and third mesh points are higher than those calculated from the formulas. This occurrence was an oversight on the part of the modulers and should not be construed as the recommended procedures.

3.6 Description of the Output

This section describes the normal and optional outputs from the meteorological module. In addition to the listing of the input deck produced by PREDAT, the METMOD program generates the normal outputs shown in Figures 3-4 through 3-11, and the optional outputs shown in Figures 3-12 through 3-15.

Figure 3-4 shows the listing for verification of air trajectory calculation inputs. These include the date, start time and location, the trajectory direction, duration, time increment, and the interpolation parameters discussed in Section 3.2. On the same page, the program prints the area source grid description and start point in both global (UTM) and local coordinates. It should be pointed out that the LOCATION OF THE INITIAL POINT will differ from that listed as the START POINT when the program is used to generate backward trajectories. This occurs because backward trajectory nodes are computed and put in chronological order between the two printings.

The trajectory data output by the program is shown in Figure 3-5. Each line of the output describes a trajectory node's time, (I, J) indices on the grid, local coordinates, and the wind velocity and direction used to reach the next trajectory node. The wind velocity is listed in the code's units of kilometers per hour. The direction is indicated by the polar angle (THETA) of the wind vector, measured in degrees counterclockwise from the x-axis. It should be pointed out that this convention for wind vector direction must be used when the user inputs an externally computed trajectory. Each trajectory node in Figure 3-5 has an alphabetic symbol associated with it. These symbols are used in the printer-plot of the trajectory, as shown in Figure 3-6. The nodes are plotted in the local coordinate system with the chronology of the nodes indicated by the alphabetic sequence of the symbols.

Next, the program lists (for several pages) the schedule of grid square crossing times, locations, and (I, J) indices on the fixed-size-grid-square grid. A sample of this output is shown in Figure 3-7. When the variable-size grid square option is used, a schedule of crossing times, locations, and single grid square identifiers is printed, as shown in Figure 3-8. It should be noted that the crossing time is the time at which the air parcel enters the grid square listed. Also, it is worth noting that the sample problem's trajectory is one which starts and ends outside of the emissions grid.

The next page of output contains regional averages of several meteorological parameters calculated for the St. Louis application of the model. Figure 3-9 shows the listing which includes the averaging times, the ultraviolet radiation data, temperature, temperature gradient in the

```
ERT METEORELOGICAL MODULE     --     ST. LOUIS     --     6/29/76


AIR TRAJECTORY CALCULATION DATA

   START DATE AND LOCAL TIME AT INITIAL POINT         760629    600 HRS (STANDARD TIME)

   LOCATION OF INITIAL POINT                             .44    -5.95   (NOT AT A MEASUREMENT STATION)

   PRESCRIBED STARTING AZIMUTH AND VELOCITY (OPTIONAL)

   DIRECTION OF TRAJECTORY DEVELOPMENT                FORWARD

   DESIRED TRAJECTORY DURATION                         11.0 HRS

   TRAJECTORY SEGMENT LENGTH                            1.0 HRS

   MEASUREMENT INTERPOLATION SCHEME                   3  CLOSEST STATIONS WITH  1/R**2  WEIGHTING




GRID REFERENCE DATA

   (KILOMETERS)               UTM COORDINATES          LOCAL COORDINATES
                                X          Y              X         Y

   NORTH WEST CORNER          680.00    4330.00          0.00    100.00

   NORTH EAST CORNER          780.00    4330.00        100.00    100.00

   SOUTH EAST CORNER          780.00    4230.00        100.00      0.00

   SOUTH WEST CORNER          680.00    4230.00          0.00      0.00

   START POINT                680.44    4224.05           .44     -5.95
```

Figure 3-4

```
ERT METEORELOGICAL MODULE      --      ST. LOUIS      --      6/29/76

TRAJECTORY DATA

  SYMBOL   T(MIN)      I      J      X(KM)      Y(KM)    V(KM/HR)    THETA

     A     360.00      1     -4        .44      -5.95      11.09     66.27

     B     420.00      5      5       4.91       4.21       8.42     62.38

     C     480.00      9     12       8.81      11.67      16.93     45.98

     D     540.00     21     24      20.58      23.84      17.18     21.53

     E     600.00     37     31      36.56      30.15      15.40     29.04

     F     660.00     51     38      50.03      37.63      15.38     25.01

     G     720.00     64     45      63.97      44.13      21.63     42.26

     H     780.00     80     59      79.97      58.67      18.18     46.72

     I     840.00     93     72      92.43      71.91      19.68     30.47

     J     900.00    110     82     109.40      81.89      21.53     28.28

     K     960.00    129     93     128.35      92.09      24.82     15.54
```

Figure 3-5

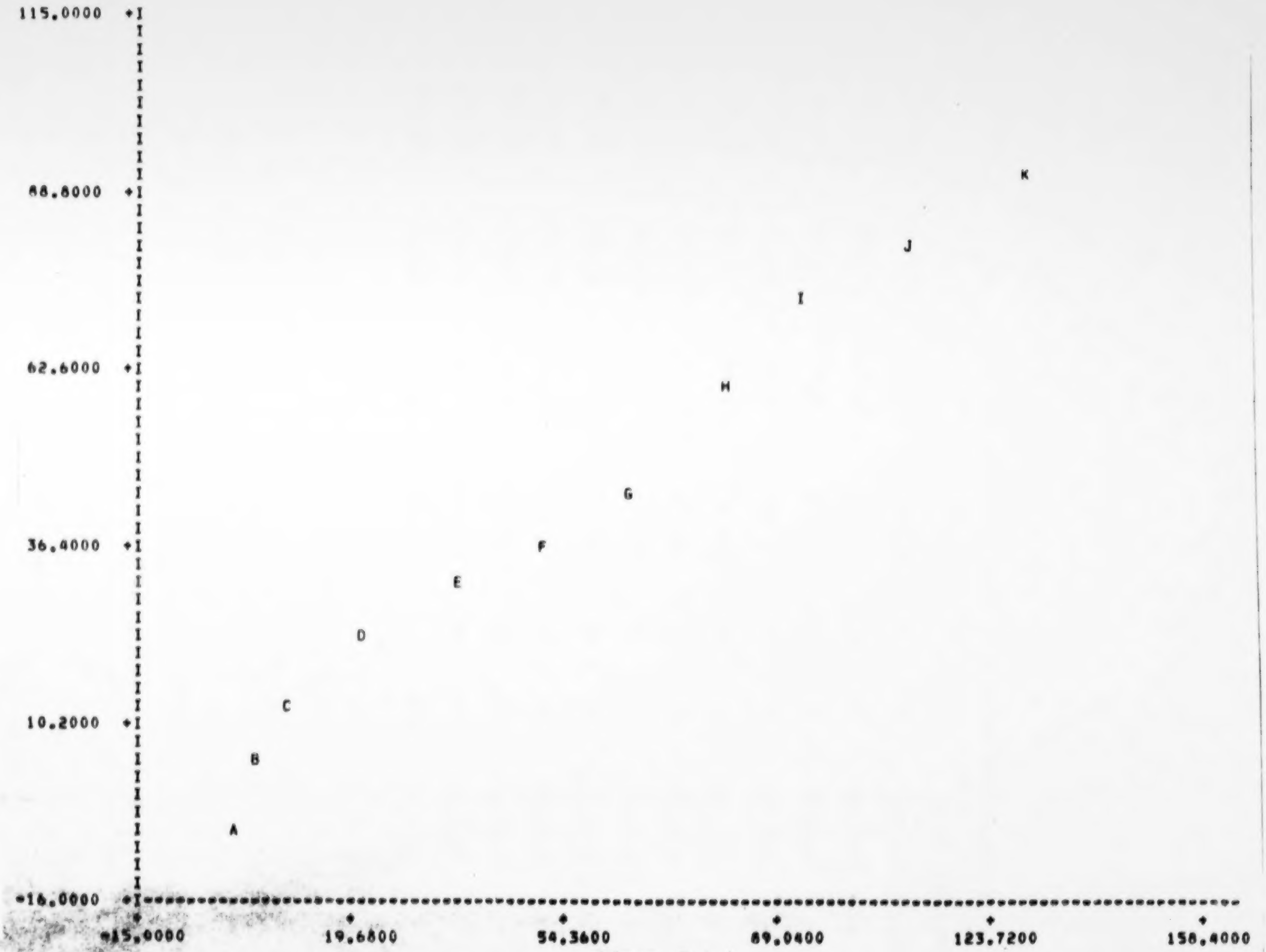

115.0000
88.8000
62.6000
36.4000
10.2000
-16.0000
-15.0000
19.6800
54.3600
89.0400
123.7200
158.4000
A
B
C
D
E
F
G
H
I
J
K

Figure 3-6

```
ERT METEORELOGICAL MODULE      --       ST. LOUIS       --

                  UTM          UTM
  TC(MIN)       XC(KM)       YC(KM)      IC      JC

   360.00       680.44      4224.05       1      -4

   367.47       681.00      4225.31       2      -3

   371.52       681.30      4226.00       2      -2

   380.91       682.00      4227.59       3      -1

   383.34       682.18      4228.00       3       0

   394.35       683.00      4229.86       4       1

   395.15       683.06      4230.00       4       1

   401.06       683.50      4231.00       4       2

   406.97       683.94      4232.00       4       3

   407.79       684.00      4232.14       5       3

   412.88       684.38      4233.00       5       4

   418.79       684.82      4234.00       5       5

   421.40       685.00      4234.38       6       5

   426.39       685.32      4235.00       6       6

   434.42       685.85      4236.00       6       7

   436.77       686.00      4236.29       7       7

   442.46       686.37      4237.00       7       8

   450.50       686.89      4238.00       7       9

   452.13       687.00      4238.20       8       9

   458.54       687.42      4239.00       8      10

   466.58       687.94      4240.00       8      11

   467.50       688.00      4240.11       9      11

   474.61       688.46      4241.00       9      12

   480.95       689.00      4241.86      10      12

   481.63       689.13      4242.00      10      13
```

Figure 3-7

```
ERT METEORELOGICAL MODULE      --      ST. LOUIS      --
 TIME (MIN)           UTM-X     UTM-Y       GRID SQUARE ID
   360.00             680.44   4224.05            0
   394.35             683.00   4229.86           64
   421.40             685.00   4234.38           72
   426.39             685.32   4235.00           65
   486.05             690.00   4242.90           81
   496.41             692.03   4245.00           82
   511.56             695.00   4248.07           99
   521.06             696.86   4250.00         2028
   537.06             700.00   4253.25          129
   551.01             703.51   4255.00          130
   556.60             705.00   4255.59         2039
   564.11             707.00   4256.38         2048
   570.03             708.58   4257.00         2049
   571.62             709.00   4257.17         2058
   579.12             711.00   4257.96         2066
   586.63             713.00   4258.74         2076
   589.06             713.65   4259.00         2077
   594.14             715.00   4259.53         2093
   598.58             716.18   4260.00         2094
   601.95             717.00   4260.39         2115
   610.86             719.00   4261.50         2137
   614.85             719.89   4262.00         2138
   619.77             721.00   4262.61         2159
   628.68             723.00   4263.72         2178
   630.89             723.50   4264.00         2179
```

Figure 3-8

```
ERT METEORELOGICAL MODULE     --     ST. LOUIS     --     6/29/76

        ST. LOUIS     YEAR =  1976     DAY = 181

        REGIONAL HOURLY AVERAGES

        HOUR             ULTRAVIOLET RADIATION     TEMPERATURE    TEMPERATURE GRADIENT     CLEARNESS RATIO
                            (LANGLEYS/MIN)           (DEG C)         (DEG C/METER)

           0 -   59           0.                      21.35              .0099                 1.000

         100 -  159           0.                      21.01              .0178                 1.000

         200 -  259           0.                      20.40              .0125                 1.000

         300 -  359           0.                      20.25              .0037                 1.000

         400 -  459           0.                      20.31             -.0055                 1.000

         500 -  559         -2.994E-03                20.85             -.0099                 1.000

         600 -  659          1.095E-02                22.05             -.0203                  .549

         700 -  759          2.552E-02                23.46             -.0274                  .703

         800 -  859          4.018E-02                25.33             -.0364                  .758

         900 -  959          5.035E-02                26.72             -.0359                  .725

        1000 - 1059          5.786E-02                28.20             -.0366                  .695

        1100 - 1159          4.903E-02                28.69             -.0330                  .545

        1200 - 1259          5.129E-02                28.32             -.0339                  .568

        1300 - 1359          8.542E-02                28.85             -.0432                 1.000

        1400 - 1459          5.227E-02                28.76             -.0328                  .727

        1500 - 1559          5.408E-02                28.95             -.0332                  .975

        1600 - 1659          3.900E-02                29.46             -.0301                 1.000

        1700 - 1759          1.724E-02                29.24             -.0206                  .772

        1800 - 1859          1.912E-03                28.08             -.0114                  .296

        1900 - 1959           0.                      25.63             -.0077                 1.000

        2000 - 2059           0.                      23.90             -.0056                 1.000

        2100 - 2159           0.                      22.60             -.0018                 1.000

        2200 - 2259           0.                      21.45              .0018                 1.000

        2300 - 2359           0.                      20.36              .0031                 1.000
```

Figure 3-9

surface layer, and the sky clearness ratios. The clearness ratios are computed as the ratio of the measured ultraviolet radiation to the expected clear-sky ultraviolet radiation for that day and hour.

The next section of output lists the interpolated values of the parameters which have been interpolated along the trajectory. An example of this output for the sample problem is illustrated in Figure 3-10. This output format is repeated for several pages with data for all trajectory node times.

The last page of the normal output, as shown in Figure 3-11, is generated by the eddy diffusivity submodule. It lists the vertical mesh point elevations and the temporal schedules of surface temperature, atmospheric stability class, and average diffusion coefficients as a function of height. The program outputs the stability class data, using the values 1 through 6, which correspond to the Pasquill-Gifford classes A (unstable) through F (very stable).

If the user has specified KPSTAT equal to YES in the input data deck, the program lists the station names and coordinates as shown in Figure 3-12. The x and y coordinates are listed in the local coordinate system even though the user has internally specified them (in SETIN) in global coordinates.

When the user elects the option for extra trajectory output (KXTRA = YES), the program provides a listing of which the station's data are used in the interpolation of wind, temperature, and air quality data along the trajectory. Figure 3-13 shows the extra output from the wind interpolation section of the program. It lists the time, location, interpolated wind speed and azimuth, and the names of the stations from which data were used. When this option is selected, the normal output of temperature and air quality data is superceded by that shown in Figure 3-14. For each trajectory node, the listing includes the qualifying station names and corresponding distances to the stations from the node. Then, for each parameter, the station data and names (in parentheses) are listed, as well as the interpolated value of the parameter. Note that the program substitutes NONE for the station name when there are no stations with valid data.

If the user selects the option for extra output from the eddy diffusivity submodule (KZPRTX = YES), a listing similar to Figure 3-15

```
ERT METEORELOGICAL MODULE     --      ST. LOUIS      --
  TIME = 1000     DATE = 760629
  VARIABLE      INTERPOLATED VALUE
     TEMP               2.7672E+01
     03                 4.3641E-02
     CO                 2.1552E-01
     CH4                1.4504E+00
     THC                1.4733E+00
     NO                 3.3445E-03
     NO2                1.1162E-02
     NOX                1.4244E-02
     TS                 1.6972E-02
     SO2                2.5000E-03

  TIME = 1100     DATE = 760629
  VARIABLE      INTERPOLATED VALUE
     TEMP               2.6457E+01
     03                 4.4532E-02
     CO                 3.6233E-01
     CH4                1.3855E+00
     THC                1.4465E+00
     NO                 2.8362E-03
     NO2                6.3903E-03
     NOX                8.1233E-03
     TS                 2.9586E-03
     SO2                2.5000E-03
```

Figure 3-10

```
ERT METEORELOGICAL MODULE     --     ST. LOUIS     --     6/29/76

     AVERAGE DIFFUSION COEFFICIENTS FOR THE CHEMICAL/DIFFUSION MODULE

     MESH POINT ELEVATIONS                        0       120       300       600      1200     METERS

     TIME    SURFACE      STABILITY        KZ1       KZ2       KZ3       KZ4       KZ5       KZ6
           TEMPERATURE      CLASS                            (M**2/MIN)

      600     21.5            2              9         9       236       327       126       126
      700     22.4            2             41        41       166       140        27        27
      800     24.4            2            487       487       389       842       507       507
      900     26.5            2           1486      1486       550       867       527       527
     1000     27.7            2           1782      1782      6202      9845      1767      1767
     1100     26.5            5            137       137       738      1495       363       363
     1200     28.5            2           3099      3099     11193     18901      6756      6756
     1300     29.6            2           2136      2136      7654     13583     10764     10764
     1400     29.4            2           1701      1701      7383     13520     10833     10833
     1500     29.3            3           1403      1403      5940     13019     10898     10898
     1600     29.7            2            104       104      5353      7509      2520      2520
```

Figure 3-11

```
ERT METEORELOGICAL MODULE     --      ST. LOUIS

          STATIONS AND COORDINATES

          RAMS 101     64.183     49.862
          RAMS 102     62.518     56.045
          RAMS 103     67.588     52.467
          RAMS 104     67.312     47.304
          RAMS 105     63.706     46.453
          RAMS 106     58.660     47.566
          RAMS 107     60.179     52.610
          RAMS 108     68.407     61.102
          RAMS 109     75.802     49.886
          RAMS 110     67.209     42.826
          RAMS 111     58.812     42.479
          RAMS 112     53.938     50.913
          RAMS 113     57.738     59.820
          RAMS 114     64.320     67.456
          RAMS 115     77.111     67.799
          RAMS 116     82.777     60.083
          RAMS 117     80.560     42.818
          RAMS 118     63.065     33.256
          RAMS 119     49.759     40.547
          RAMS 120     43.079     55.909
          RAMS 121     52.414     72.376
          RAMS 122     61.631     99.223
          RAMS 123     97.320     56.378
          RAMS 124     69.275      6.537
          RAMS 125     17.445     52.240
```

Figure 3-12

```
OLD TIME - DATE          X      Y          NEW TIME - DATE        VELOCITY AZIMUTH     STATIONS

     600     760629       .44  -5.95           600     760629       11.09    66.27     125    119    124
     700     760629      4.91   4.21           700     760629        8.42    62.38     125    119    120
     800     760629      8.81  11.67           800     760629       16.93    45.98     125    119    120
     900     760629     20.58  23.84           900     760629       17.18    21.53     125    119    120
    1000     760629     36.56  30.15          1000     760629       15.40    29.04     119    111    120
    1100     760629     50.03  37.63          1100     760629       15.38    25.01     119    111    106
    1200     760629     63.97  44.13          1200     760629       21.63    42.26     105    110    104
    1300     760629     79.97  58.67          1300     760629       18.18    46.72     116    115    109
    1400     760629     92.43  71.91          1400     760629       19.68    30.47     116    115    123
    1500     760629    109.40  81.89          1500     760629       21.53    28.28     123    116    115
    1600     760629    128.35  92.09          1600     760629       24.82    15.54     123    116    115
```

Figure 3-13

```
ERT METEORELOGICAL MODULE      --      ST. LOUIS       --      6/29/76

 TIME =  1000      STATION NUMBERS              119    111    120    118    112
                                                106    125    105    107    110
                                                101    104    113    102    103
                                                124    109    108    121    117
                                                114    116    115    123

 DATE = 760629     DISTANCES (KM)             16.80  25.44  26.57  26.68  27.08
                                              28.14  29.21  31.66  32.59  33.17
                                              33.93  35.21  36.45  36.66  38.22
                                              40.34  43.92  44.41  45.10  45.79
                                              46.50  55.06  55.33  66.18

  VARIABLES      X1     (STAT)        X2     (STAT)         X3    (STAT)   X-INTERPOLATED
   TEMP      2.70E+01  ( 119)    2.83E+01  ( 111)    2.86E+01  ( 120)     2.7672E+01
   O3        4.23E-02  ( 119)    4.29E-02  ( 111)    4.79E-02  ( 120)     4.3641E-02
   CO        2.29E-01  ( 119)    3.27E-01  ( 111)    5.89E-02  ( 120)     2.1552E-01
   CH4       1.38E+00  ( 119)    1.50E+00  ( 111)    1.57E+00  ( 120)     1.4504E+00
   THC       1.43E+00  ( 119)    1.46E+00  ( 111)    1.59E+00  ( 120)     1.4733E+00
   NO        3.88E-03  ( 119)    2.89E-03  ( 111)    2.50E-03  ( 120)     3.3445E-03
   NO2       1.14E-02  ( 119)    1.21E-02  ( 111)    9.50E-03  ( 120)     1.1162E-02
   NOX       1.49E-02  ( 119)    1.49E-02  ( 111)    1.19E-02  ( 120)     1.4244E-02
   TS        2.26E-02  ( 119)    1.73E-02  ( 111)    2.50E-03  ( 120)     1.6972E-02
   SO2       2.50E-03  ( 120)    2.50E-03  ( 108)   -1.00E+00  (NONE)     2.5000E-03
 TIME =  1100      STATION NUMBERS              119    111    106    118    112
                                                105    110    107    101    120
                                                104    102    103    113    109
                                                108    117    114    121    125
                                                124    116    115    123

 DATE = 760629     DISTANCES (KM)              2.93  10.03  13.16  13.75  13.85
                                              16.28  17.95  18.10  18.71  19.56
                                              19.81  22.25  22.99  23.49  28.54
                                              29.81  30.97  33.08  34.83  35.71
                                              36.57  39.71  40.54  50.87

  VARIABLES      X1     (STAT)        X2     (STAT)         X3    (STAT)   X-INTERPOLATED
   TEMP      2.63E+01  ( 119)    2.66E+01  ( 111)    2.85E+01  ( 106)     2.6457E+01
   O3        4.49E-02  ( 119)    3.88E-02  ( 111)    4.71E-02  ( 106)     4.4532E-02
   CO        3.65E-01  ( 119)    3.73E-01  ( 111)    2.92E-01  ( 106)     3.6233E-01
   CH4       1.37E+00  ( 119)    1.52E+00  ( 111)    1.55E+00  ( 106)     1.3855E+00
   THC       1.44E+00  ( 119)    1.44E+00  ( 111)    1.61E+00  ( 106)     1.4465E+00
   NO        2.88E-03  ( 119)    2.50E-03  ( 111)    2.50E-03  ( 106)     2.8362E-03
   NO2       5.64E-03  ( 119)    1.10E-02  ( 111)    1.37E-02  ( 106)     6.3903E-03
   NOX       7.27E-03  ( 119)    1.35E-02  ( 111)    1.62E-02  ( 106)     8.1233E-03
   TS        2.50E-03  ( 119)    8.57E-03  ( 111)    2.50E-03  ( 118)     2.9586E-03
   SO2       2.50E-03  ( 108)   -1.00E+00  (NONE)   -1.00E+00  (NONE)     2.5000E-03
```

Figure 3-14

```
ERT METEORELOGICAL MODULE     --     ST. LOUIS     --     6/29/76

          TIME    1300 LST

VERTICAL EDDY DIFFUSIVITY EVERY  20 METERS

    HEIGHT     KZ         HEIGHT     KZ         HEIGHT     KZ         HEIGHT     KZ         HEIGHT     KZ

       20     490.0         620  15794.4         1220      6.0         1820      6.0         2420      6.0
       40    1151.7         640  15872.5         1240      6.0         1840      6.0         2440      6.0
       60    1895.8         660  15921.7         1260      6.0         1860      6.0         2460      6.0
       80    3151.0         680  15941.4         1280      6.0         1880      6.0         2480      6.0
      100    3992.9         700  15931.0         1300      6.0         1900      6.0         2500      6.0
      120    4804.4         720  15889.6         1320      6.0         1920      6.0         2520      6.0
      140    5585.7         740  15816.4         1340      6.0         1940      6.0         2540      6.0
      160    6337.2         760  15709.9         1360      6.0         1960      6.0         2560      6.0
      180    7059.0         780  15568.7         1380      6.0         1980      6.0         2580      6.0
      200    7751.3         800  15390.9         1400      6.0         2000     27.1         2600      6.0
      220    8414.5         820  15174.1         1420      6.0         2020     17.1         2620      6.0
      240    9048.7         840  14915.4         1440      6.0         2040      9.1         2640      6.0
      260    9654.2         860  14611.0         1460      6.0         2060      6.0         2660      6.0
      280   10231.1         880  14255.8         1480      6.0         2080      6.0         2680      6.0
      300   10779.6         900  13843.4         1500      6.0         2100      6.0         2700      6.0
      320   11300.0         920  13364.7         1520      6.0         2120      6.0         2720      6.0
      340   11792.4         940  12806.9         1540      6.0         2140      6.0         2740      6.0
      360   12256.9         960  12151.1         1560      6.0         2160      6.0         2760      6.0
      380   12693.6         980  11367.1         1580      6.0         2180      6.0         2780      6.0
      400   13102.8        1000  10402.2         1600      6.0         2200      6.0         2800      6.0
      420   13484.5        1020   9148.5         1620      6.0         2220      6.0         2820      6.0
      440   13838.8        1040   7307.2         1640      6.0         2240      6.0         2840      6.0
      460   14165.8        1060      6.0         1660      6.0         2260      6.0         2860      6.0
      480   14465.4        1080      6.0         1680      6.0         2280      6.0         2880      6.0
      500   14737.8        1100      6.0         1700      6.0         2300      6.0         2900      6.0
      520   14982.8        1120      6.0         1720      6.0         2320      6.0         2920      6.0
      540   15200.5        1140      6.0         1740      6.0         2340      6.0         2940      6.0
      560   15390.7        1160      6.0         1760      6.0         2360      6.0         2960      6.0
      580   15553.2        1180      6.0         1780      6.0         2380      6.0         2980      6.0
      600   15687.8        1200      6.0         1800      6.0         2400      6.0         3000      6.0

    AVERAGE DIFFUSION COEFFICIENTS FOR THE CHEMICAL/DIFFUSION MODULE

    TIME      SURFACE      STABILITY       KZ1      KZ2       KZ3       KZ4       KZ5       KZ6
            TEMPERATURE      CLASS                          (M**2/MIN)

    1300       29.6            2          2136     2136      7654     13583     10764     10764
```

Figure 3-15

is generated for each hour. This figure shows the detailed (internal) eddy diffusivity coefficients as a function of elevation. Each hour's vertical profile is printed prior to the hour's average diffusion coefficients.

When the user selects the option for punched output (KPWIND = YES), the user obtains card images punched on logical unit number 1 with the following data:

1) the page title
2) trajectory start location in global coordinates
3) STANDARD or DAYLIGHT time card
4) trajectory node times, wind velocities and directions
5) trajectory grid square crossing times, coordinates, and either (I, J) indices or single identifiers
6) schedule of sky clearness ratios
7) schedule of surface temperatures
8) schedule of average diffusivity coefficients
9) schedule of atmospheric stability classes.

These card-images are punched in the order listed above with all references to time made on the 0-1440 minutes/day clock.

3.7 Internally Specified Data

The METMOD program requires internal specification of certain parameters specific to a particular modeling region. These parameters are designated using either FORTRAN DATA statements or FORTRAN replacement statements. Table 3-5 lists the code variable names, their functions, and the subroutines in which they are specified. In addition, the FORTRAN source listing contains notes to the user where internal specification of data is required.

3.8 Surface Roughness Lengths

The eddy diffusivity submodule requires the specification of surface roughness length parameters. Figure 3-16 illustrates the surface roughness profile used in the model. This profile has been designed for

TABLE 3-5

METEOROLOGICAL MODULE INTERNALLY SPECIFIED DATA

Variable Name	Function	Subroutine
NUMSTA	Number of measurement stations (wind, temperature, radiation, etc.)	SETIN
ISTANS	The measurement station names. A vector dimensioned 2 x NUMSTA which contains two 4-character Hollerith words for each station. The second word for each name must be unique.	SETIN
SSTAN	The measurement station locations. A vector dimensioned 2 x NUMSTA which contains the global (UTM) horizontal and vertical coordinates of the stations. (kilometers)	SETIN
RLAT	Latitude of the approximate center of the modeling region. Used in solar zenith angle computation. (degrees)	SETIN
RLONG	Longitude of approximate center of the modeling region. Used in solar zenith angle computation. (degrees)	SETIN
TMZONE	Time zone of the modeling region. Used in the solar zenith angle computation. Use 6. for St. Louis, which represents the difference between local time and Greenwich, England time.	SETIN
NUMBAS	Number of geographic terrain barriers specified. (max = 20)	BARIER
BAS	The global (UTM) end point coordinates of terrain barrier in the modeling region. A vector dimensioned 4 x NUMBAS which contains the x_1, y_1, x_2, y_2 coordinates of a line segment representing a physical barrier to the wind field. (kilometers)	BARIER
BUFZON	The width of a buffer zone outside of emissions grid. Backward trajectories are not permitted to back off the grid more than this distance. (kilometers)	EDGE
ZW	Elevation of wind measurement instrument above ground. (meters)	EDDY

TABLE 3-5 (CONTINUED)

Variable Name	Function	Subroutine
UTMXUR, UTMYUR	Coordinates (x,y) of the center of the urban roughness length region. (UTM = kilometers)	RUFNES
ZOURBN	The urban surface roughness length. (meters)	RUFNES
ZORURL	The rural surface roughness length. (meters)	RUFNES
RADURB	Radius of urban surface roughness length region. (kilometers)	RUFNES
RADSUB	Radius to rural surface roughness length region. (kilometers)	RUFNES
NSTMAX	The maximum number of measurement stations for which the program's arrays are dimensioned.	METIN
NOSPEC	Number of variables stored in the CON array for interpolation along the trajectory. (max = 10)	METIN
CONAM	A vector of 4-character Hollerith words which contains the names of the variables in the CON array.	METIN
ITGRID	A Hollerith variable equal to YES when the variable size grid square feature of the program is to be used.	BLOCK DATA
VGRID	A two-dimensional array which contains the variable size grid square identifiers corresponding to a grid with uniform size grid squares. (See Section 3.4)	BLOCK DATA

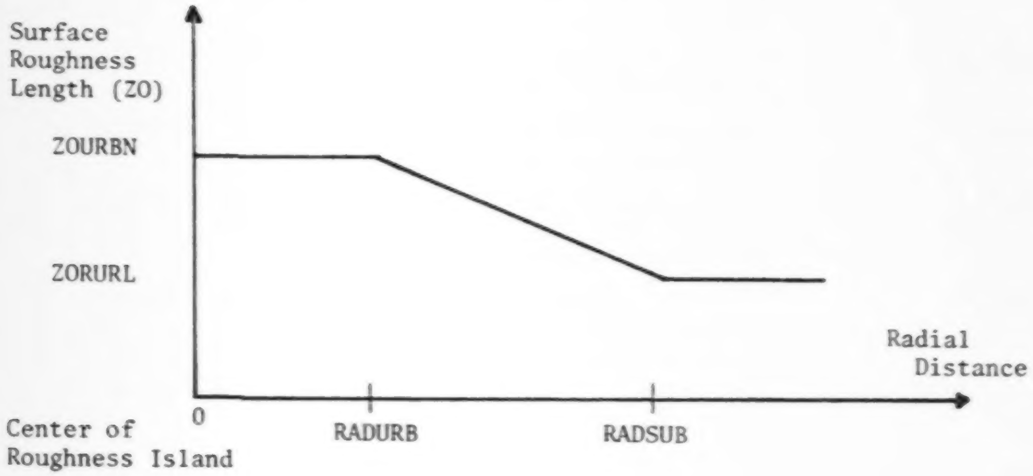


Figure 3-16 Surface Roughness Length Profile

modeling a region with a surface roughness island like St. Louis. The program assumes there is a circular subregion with a district surface roughness length ZOURBN and an outlying region with surface roughness length ZORURL. The user specifies these two roughness lengths in meters, the global x and y coordinates of the center of the subregion, and the radii of the inner (urban) and outer (suburban) regions, RADURB and RADSUB, respectively. The program assumes that the surface roughness varies linearly with radial distance in the region between the inner and outer radii. These parameters are specified in subroutine RUFNES, as described in Section 3.8.

For applications without a surface roughness island, the user can specify ZOURBN equal to ZORURL. Similarly, the user may have an application where an inverted profile (ZOURBN < ZORURL) is appropriate, such as a heavily forested outlying region. In any case, the user must choose RADURB not equal to RADSUB for proper operation of the code.

3.9 Printer-Plot Boundaries

The printer-plot boundaries specified in the input deck may focus on smaller areas than the complete modeling region. The parameters XL, XR, YB, and YT are, respectively, the local x-coordinates of the left and right (east) boundaries of the plotted areas, and the local y-coordinates of the bottom and top (north) boundaries of the plotted area. These parameters are input in kilometers.

Because of the constraints involved in using a line printer to produce a plot (fixed at 51 horizontal lines and 101 characters in this case), certain rules of thumb should be followed in selecting the plot boundaries in order to obtain undistorted representations. Since character spaces are 0.6 times as large as line spaces, true-to-scale plots can be obtained only by maintaining a ratio of .84:1 between the character position scale and the print line scale. Thus, the user is instructed to choose XR, XL, YB, and YT such that

$$\frac{YT - YB}{XR - XL} \simeq .84$$

to obtain undistorted plots of the trajectories.

6) CLOSE STATIONS: If a station is within a minimum distance (RMIN) from a particular trajectory node, the program uses only the data from this station. When there are no stations within the minimum distance, or when a station within the minimum distance has no data available, the program searches for data from up to three neighboring stations. It uses the measurements from the neighboring close stations, provided they are located within the maximum distance, RMAX, from the node.

7) EDGE TEST: Development of a backward air trajectory may be terminated early if its path takes it out side the modeling region. (Subroutine EDGE)

Having established this background of terms, we are ready to describe the calculation sequences for forward and reverse trajectory development. Forward trajectory development consists of cyclic execution of the following steps:

1) Search the wind data of the stations in order of increasing distance from the trajectory point until the desired number of close stations having valid data are found, or RMAX is reached. These operations are accomplished in subroutine GETAZV. Lacking any close stations, the trajectory calculation is terminated.

2) Combine the n wind speed/azimuth pairs found into an average speed and azimuth using the interpolation formula

$$\overline{S} = \sum_{i=1}^{n} \frac{\frac{1}{R_i^k} S_i}{\sum_{i=1}^{n} \frac{1}{R_i^k}} \qquad (3\text{-}1)$$

where i = station index

n = number of close stations to be used
($1 \leq n \leq 3$, as specified by the user)

k = 1 or 2 as specified by the user

R_i = distance from trajectory point to i^th station

Preceding page blank

S_i = speed or angle measurement from i^{th} station
$\overline{S}$ = weighted average speed or azimuth

This calculation is performed in subroutine AZVDIS.

3) Integrate the trajectory one time step using Euler's method (i.e., using the current position and velocity to extrapolate the trajectory to the next node in a single step).

4) Unless the trajectory is complete, and provided the barrier and edge tests permit, start again at step (1) using the time for the new node.

Reverse trajectory development is somewhat more complicated, but does utilize many of the calculations needed in the forward case. Reverse trajectories are calculated in the following manner:

1) Search the station measurement data using the current trajectory point, using the time that will be appropriate at the next node (that being one time step earlier).

2) Combine the measurements.

3) Extrapolate the trajectory back one time step, using the current position and the velocity calculated for one time step earlier, to obtain an estimate of the next earlier node position (see Figure 3-17a).

4) Using the estimated node position, search the station data with the same time for a new set of close station data.

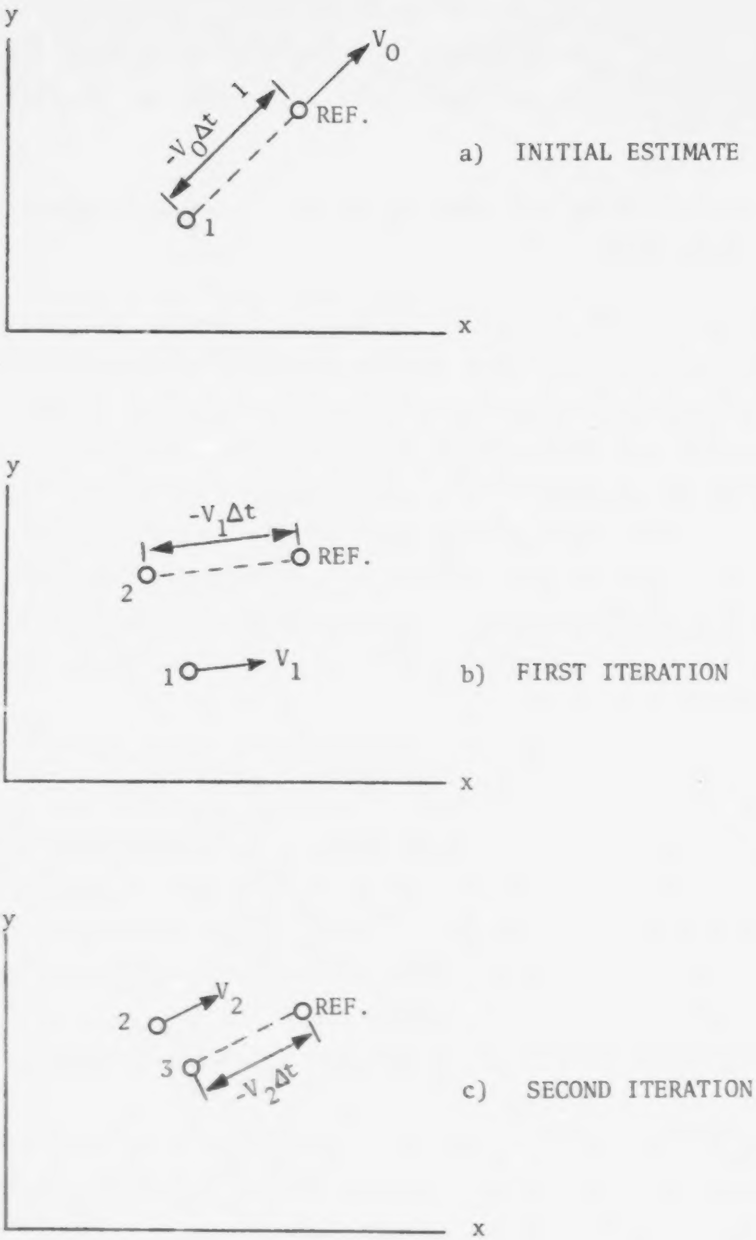
5) Combine the measurements.

6) Starting from the reference node, reestimate the position of the next earlier node by extrapolating back one time step with the new wind vector (see Figure 3-17b).

7) Repeat steps 4), 5), and 6) as desired (program currently does three iterations, see Figure 3-17c).

8) Barrier and edge test permitting, establish the last estimate of the next earlier node as the new reference node.

9) Record the last calculated wind vector with the time appropriate to the new node.



a) INITIAL ESTIMATE

b) FIRST ITERATION

c) SECOND ITERATION

Figure 3-17 Reverse Trajectory Development

10) Return to step 1), unless the desired trajectory has been completed.

When a reverse trajectory has been completed, subroutine WINDY reorders the table of times and wind vectors, placing them in chronological order, and records the final calculated point as the trajectory start point.

3.10.2 Interpolation of Other Atmospheric Parameters Along the Trajectory

The code calculates the surface temperature and air quality data at trajectory nodes in a manner very similar to that used to obtain wind vectors. The program searches the measurement records of close stations for each parameter and interpolates the valid data (from up to three stations) using the Equation (3-1). The principal difference between the screening of these data and the wind data is that these data are not subjected to the angle test or barrier test. The rules for extrapolation of data and determination of close stations are identical to those described in the previous section. The calculations are made in subroutine GETCON.

3.10.3 Calculation of Eddy Diffusivity Coefficients

The computation of vertical eddy diffusivity coefficients (K_z) occurs after the generation of air trajectory and the interpolation of atmospheric parameters. The trajectory wind speeds, trajectory node locations, and the interpolated surface temperatures are passed to subroutine EDDY, which controls the generation of the K_z schedules. This subroutine calls KZDATA to read the vertical temperature data and the vertical mesh point elevations.

This subprogram's computation cycle employs an outer loop for each vertical temperature profile and a nested (inner) loop for calculation of hourly K_z profiles between sounding release times.

For each cycle through the outer loop, the program partitions the vertical temperature structure into (up to 30) layers with distinct lapse rates. Temperature data points which indicate very minor variations in the lapse rate are removed from the vertical profile by subroutine SMOOTH.

Given the edited vertical temperatures and other data mentioned above, the program proceeds to the inner loop. The first computation within the loop involves updating the vertical temperature profile if the soundings release time precedes the trajectory start time (or current time). The update is performed using the incremental difference between the current hour's surface temperature and that of the previous hour or the sounding, whichever is closer. The lapse rate used to update the sounding is determined from the following rules:

1) If the increment in surface temperature is positive, the program uses the greater of the lapse rates in the previous vertical sounding, the next vertical sounding, or the dry adiabetic lapse rate.
2) If the increment in surface temperature is negative, the program uses the lapse rate indicated by the surface temperature and first elevated temperature.

Once the sounding has been updated, the program calculates the surface roughness length for the particular trajectory location. Now the code has all the information required to calculate the current hour's K_z profile. The stability of each atmospheric layer with a distinct lapse rate is determined. In addition, the atmospheric stability class of the surface layer is computed for use in the emissions module. Within each layer, the K_z values are calculated at either 10 or 20 meter vertical increments, using different K_z formulations for within and above the surface layer. Once this detailed K_z profile is computed, the program calculates a set of average diffusion coefficients (averaged between the mesh points) for use in the chemical-diffusion model.

At this point, the program remains in the inner loop (i.e., updating the sounding and calculating K_z profiles) until a new sounding time is reached. Then, control is passed to the outer loop, where the entire process is repeated until the final trajectory time is reached.

3.11 Program Modifications

3.11.1 Changing the Number of Measurement Stations

The variable NUMSTA located in common block /WDATA/ is the number of stations recognized by the program. Many arrays in the program have dimensions which must be equal or larger than NUMSTA. Currently, NUMSTA and these dimensions are set equal to 25 for the St. Louis application of the model. If the user desires to use N stations (N > 25), the following arrays should be redimensioned:

In all subroutines with common blocks:

COMMON/AIRQAL/ .... CON(24,N,10) ....

COMMON/WDATA/ .... SSTAN(2,N), ISTANS(2,N) ...

COMMON/WINFLD/ .... AWDATA(52,N), DISAVE(N), IDSAVE(N)

In subroutines:

GETAZV	DIS(N), IDIS(N), DDIS(N)
GETCON	IWORK(N), WORK(N)
METIN	NWDATA(52,N), X(22,N), WORK(22,N)

In addition, NUMSTA and NSTMAX should be set equal to N in subroutines SETIN and METIN, respectively. See Appendix A for a list of subroutines with the common blocks mentioned above.

3.11.2 Changing the Station Azimuth Exclusion Sector

Currently, data will not be accepted from two stations lying within 2^{o} azimuth of each other, as seen from the trajectory node being considered. The size of this exclusion angle is determined by the variable COSTST set in a DATA statement in subroutine ANGTST. COSTST should be set equal to the cosine of the desired exclusion angle. Note that since the exclusion angle is applied on both sides of the azimuth to a previously accepted station, the sector actually excluded is twice the specified angle.

4. THE EMISSIONS MODULE

This section provides instructions for proper use of the emissions module, program EMMOD. The function of the program is to process data from a region's emissions inventory to produce a tabulation of emissions along a trajectory for use in the chemical-diffusion module. The program requires regionally disaggregated emissions data for nitrogen oxides, sulfur oxides, carbon monoxide, and four classes of reactive hydrocarbon species: alkanes (paraffins), alkenes (olefins), aromatics, and aldehydes.

Program EMMOD is designed to utilize emission inventories which have been preprocessed into modeling format, a format in which the contributions of individual source types (mobile and stationary) have been summed into single hourly area source emission rates for each grid square. In addition, the program considers the major emission point sources on an individual basis. The inputs to the program include a small file of control parameters and trajectory data read from the card reader. Also included are three tape files, which contain the area source grid description; the hourly area source emission rates; and, the point source locations, hourly emission rates, and stack parameters.

Portions of the program are designed specifically for the format of the RAPS/St. Louis emissions tapes. The user should anticipate making significant modifications to the code for applications involving regions other than St. Louis. These modifications are necessary not only because of the different formats of emission data in different regions, but also in the interest of efficient processing of the data.

4.1 Control Parameters and the Input Deck

The input deck parameters will be discussed in the general order in which they are input to the code. The reader should refer to several figures and a table, as described below, to obtain a better understanding of the input data and format. Figure 4-1 illustrates the listing of the input deck produced by subroutine PREDAT for the sample problem. Table 4-1 lists the inputs by their code variable names, and describes their functions and formats. Figure 4-2 contains a flow chart of the input variable sequence and options. These aids will be helpful in formulating input decks for the program.

TABLE 4-1

EMISSIONS MODULE INPUT VARIABLE DESCRIPTION

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
1	TITLE	Label for each page of output.	(20A4)	May contain any user comments in columns 1-80.
2	NDAREA	Area source emission schedules are to be computed (YES or NO).	(40X,A4)	This is a 4-character Hollerith variable equal to YES or NO starting in column 41.
3	NDPONT	Point source emission schedules are to be computed (YES or NO).	(40X,A4)	
4	NDSUMS	Regional sums of emissions each hour are to be computed (YES or NO).	(40X,A4)	
5	IPUNCH	Outputs required as input for the chemical-diffusion module are to be punched (YES or NO).	(40X,A4)	
6	LUGRID, LUAREA, LUPONT	Logical unit number of area source grid file, area source emissions file, and point source emissions file.	(40X,I2)	(Three cards)
7	NFLXIN	Number of input emission rates.	(40X,I2)	
8	NFXOUT	Number of output emission rates.	(40X,I2)	
9	NAMIN, ADJUST	Hollerith names and conversion factors of input emission species.	(40X,A4,6X, F12.0)	(NFLXIN cards)

TABLE 4-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
10	NAMOUT	Hollerith names of output emission species	(40X,A4,6X, F10.0)	(NFXOUT cards)
11	PLENTH, PWIDTH	Length and width of the air parcel in kilometers.	(40X,F10.0)	Must correspond to the minimum grid square size. (Two cards)
12	EMABAS	Default area source emission rates for the region outside grid.	(40X,F10.0)	(NFXOUT cards)
13	TIMEHS, UTMAS, IDHS	Grid square crossing times, UTM coordinates, and corresponding grid square ID numbers.	(40X,3F10.0, I10)	(Output from METMOD)
14	UTMXOR, UTMYOR	Global x and y coordinates of local coordinate system origin in kilometers.	(40X,2F10.0)	
15	P(1,1), P(2,1)	Trajectory start point global x and y coordinates in kilometers.	(40X,2F10.0)	
16	T, V, TH	Time, wind velocity, and azimuth of trajectory nodes.	(40X,3F10.0)	Time in minutes, velocity in km/hr, and azimuth in degrees counterclockwise from east (output from METMOD).
17	NOSTAT	Number of vertical mesh points.	(40X,I10)	
18	ZEE	Elevations of vertical mesh points in air parcel in meters.	(40X,F10.0)	(NOSTAT cards)
19	TMTEMP, TEMPSF	Times and values of surface temperatures.	(40X,2F10.0)	Time in minutes and temperature in degrees centigrade.

TABLE 4-1 (CONTINUED)

Input Number	Variable Names Used In Code	Variable Function	FORTRAN Format	Remarks
20	TMSTAB, KSTABL	Times and values of atmospheric stability class.	(40X,2F10.0)	Time in minutes and stability class A-F are specified as 1-6.
21	TMXHIT, HTMIXL	Schedule of times and corresponding mixing layer heights or inversion heights, whichever is greater.	(40X,2F10.0)	
22	TERM	Last card which has the value of MORE or END.		Note special JCL is needed for multiple runs.

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
   1 ERT EMISSIONS MODULE     --     ST. LOUIS      --     6/29/76   (181)
   2 DO WE NEED AREA SOURCE EMISSIONS        YES
   3 DO WE NEED POINT SOURCE EMISSIONS       YES
   4 DO WE NEED REGIONAL SUMS                YES
   5 DO WE NEED PUNCHED OUTPUT               YES
   6 LOGICAL UNIT OF GRID DESCRIPTION        10
   7 LOGICAL UNIT OF AREA SOURCE EMISSIONS   11
   8 LOGICAL UNIT OF POINT SOURCE EMISSIONS  12
   9 NUMBER OF INPUT VARIABLES               11
  10 NUMBER OF OUTPUT VARIABLES              07
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  11 INPUT VARIABLE NAMES,ADJUSTMENT         PART      1.0
  12 2                                       SOX       15.625     FROM KG TO MOLES
  13 3                                       NOX       21.739     TO MOLES NO
  14 4                                       THC       25.0
  15 5                                       CO        35.714     TO MOLES CO
  16 6                                       DUST      1.0
  17 7                                       NRHC      1.0
  18 8                                       PARF      1.0
  19 9                                       OLEF      1.0
  20 10                                      AROM      1.0
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  21 11                                      ALDE      1.0
  22 OUTPUT VARIABLE NAMES,                  NOX
  23 2                                       PARF
  24 3                                       OLEF
  25 4                                       AROM
  26 5                                       ALDE
  27 6                                       CO
  28 7                                       SOX
  29 WIDTH OF THE AIR PARCEL    (KM)         1.0
  30 LENGTH OF THE AIR PARCEL   (KM)         1.0
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  31 DEFAULT AREA SOURCE RATES  NOX             0.0E 00     MOLES PER HOUR
  32 2                          PARF            0.0E 00
  33 3                          OLEF            0.0E 00
  34 4                          AROM            0.0E 00
  35 5                          ALDE            0.0E 00
  36 6                          CO              0.0E 00
  37 7                          SOX             0.0E 00
  38 TM,X,Y,ID  360.00    680.44    4224.05           0
  39 2          394.40    683.00    4229.87          64
  40 3          421.46    685.32    4234.39          72
```

Figure 4-1 Emissions Module Sample Problem Input Listing

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  41  4          426.39    685.32    4235.00          65
  42  5          486.07    690.00    4242.90          81
  43  6          496.41    692.03    4245.00          82
  44  7          511.58    695.00    4248.08          99
  45  8          521.06    696.86    4250.00        2028
  46  9          537.08    700.00    4253.25         129
  47  10         551.01    703.50    4255.00         130
  48  11         556.62    705.00    4255.59        2039
  49  12         564.13    707.00    4256.38        2048
  50  13         570.04    708.57    4257.00        2049

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  51  14         571.63    709.00    4257.17        2058
  52  15         579.14    711.00    4257.96        2066
  53  16         586.65    713.00    4258.75        2076
  54  17         589.06    713.64    4259.00        2077
  55  18         594.15    715.00    4259.53        2093
  56  19         598.58    716.18    4260.00        2094
  57  20         601.97    717.00    4260.39        2115
  58  21         610.88    719.00    4261.51        2137
  59  22         614.85    719.89    4262.00        2138
  60  23         619.79    721.00    4262.62        2159

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  61  24         628.70    723.00    4263.73        2178
  62  25         630.90    723.49    4264.00        2179
  63  26         637.61    725.00    4264.84        2198
  64  27         646.52    727.00    4265.95        2212
  65  28         646.94    727.09    4266.00        2213
  66  29         655.43    729.00    4267.06        2227
  67  30         663.44    730.82    4268.00        2228
  68  31         664.20    731.00    4268.08        2245
  69  32         672.81    733.00    4269.02        2255
  70  33         681.42    735.00    4269.95         706

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  71  34         681.90    735.11    4270.00         707
  72  35         685.72    736.00    4270.41         746
  73  36         690.03    737.00    4270.88         781
  74  37         691.13    737.36    4271.00         782
  75  38         694.33    738.00    4271.35         818
  76  39         698.64    739.00    4271.81         850
  77  40         700.36    739.40    4272.00         851
  78  41         702.94    740.00    4272.28         889
  79  42         707.25    741.00    4272.75         924
  80  43         709.59    741.54    4273.00         925
```

Figure 4-1 (Continued)

```
         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
   81    44            711.55    742.00   4273.21       955
   82    45            715.85    743.00   4273.68       998
   83    46            723.60    744.92   4275.00      1030
   84    47            723.89    745.00   4275.07      1066
   85    48            727.64    746.00   4275.98      1095
   86    49            727.72    746.02   4276.00      1096
   87    50            731.38    747.00   4276.89      1119
   88    51            731.85    747.12   4277.00      1120
   89    52            735.13    748.00   4277.80      1152
   90    53            735.97    748.22   4278.00      1153
         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
   91    54            738.88    749.00   4278.71      1182
   92    55            740.10    749.32   4279.00      1183
   93    56            742.63    750.00   4279.61      1219
   94    57            744.22    750.42   4280.00      1220
   95    58            746.38    751.00   4280.52      1254
   96    59            748.35    751.53   4281.00      1255
   97    60            750.13    752.00   4281.43      1283
   98    61            752.47    752.63   4282.00      1222
   99    62            753.87    753.00   4282.34      1317
  100    63            756.60    753.73   4283.00      1318
         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
  101    64            757.62    754.00   4283.25      1350
  102    65            760.72    754.83   4284.00      1351
  103    66            761.37    755.00   4284.16      2326
  104    67            764.85    755.93   4285.00      1391
  105    68            780.14    760.00   4288.70      1499
  106    69            786.01    761.22   4290.00      1500
  107    70            804.21    765.00   4294.01      1592
  108    71            808.68    765.93   4295.00      2374
  109    72            809.03    766.00   4295.08      2392
  110    73            817.74    767.61   4297.00      2393
         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
  111    74            818.65    768.00   4297.20      2404
  112    75            826.81    769.69   4299.00      2405
  113    76            828.28    770.00   4299.32      1626
  114    77            831.34    770.64   4300.00      1627
  115    78            849.08    775.00   4303.42      1642
  116    79            858.58    777.69   4305.00      1643
  117    80            866.76    780.00   4306.36         0
  118    LAST          -10.00
  119    UTM COORDINATES OF GRID ORIGIN          680.00     4230.00
  120    STARTING LOCATION UTM X & Y             680.44     4224.05
```

Figure 4-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 121  TRAJECTORY DESCRIPTIONS                 360.0     11.09     66.27
 122  B  TIME(MIN), WIND V & D (KM/HR & DEG)  420.00     8.421    62.38
 123  C                                       480.00    16.93     45.98
 124  D                                       540.00    17.18     21.53
 125  E                                       600.00    15.40     29.04
 126  F                                       660.00    15.38     25.00
 127  G                                       720.00    21.63     42.26
 128  H                                       780.00    18.18     46.72
 129  I                                       840.00    19.68     30.47
 130  J                                       900.00    21.53     28.28

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+........
 131  K                                       960.00    24.82     15.54
 132  L                                       1020.0    24.82     15.54
 133  M                                       1080.0    24.82     15.54
 134  LAST TRAJECTORY NODE (DUMMY NEGATIVE)   -100.0
 135  NUMBER OF KEMOD VERTICAL MESH POINTS     05
 136  MESH POINT ELEVATIONS (ABOVE SURFACE)    0.0        (METERS)
 137  2                                        120.0
 138  3                                        300.0
 139  4                                        600.0
 140  5                                        1200.0

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+........
 141  SURFACE TEMPERATURES                      360.     21.50
 142                                            420.     22.41
 143                                            480.     24.39
 144                                            540.     26.46
 145                                            600.     27.67
 146                                            660.     26.46
 147                                            720.     28.51
 148                                            780.     29.61
 149                                            840.     29.43
 150                                            900.     29.27

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 151                                            960.     29.66
 152                                           1020.     29.64
 153  LAST DUMMY SURFACE TEMP (NEGATIVE)       - 10.
 154  STABILITY CLASSES                       360.               2
 155    USE 1-6 FOR A-F                       660.               5
 156                                          720.               2
 157                                          900.               3
 158                                          960.               2
 159  LAST STABILITY (DUMMY NEGATIVE)         -10.
 160  MIXING LAYER DEPTHS (METERS)            360.        200.
```

Figure 4-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 161                                          420.      270.
 162                                          480.      340.
 163                                          540.      410.
 164                                          600.      490.
 165                                          660.      600.
 166                                          720.      710.
 167                                          780.      820.
 168                                          840.      930.
 169                                          900.     1040.
 170                                          960.     1080.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 171  LAST MIXING LAYER DEPTH (NEGATIVE)      -10.
 172  END
```

Figure 4-1 (Continued)

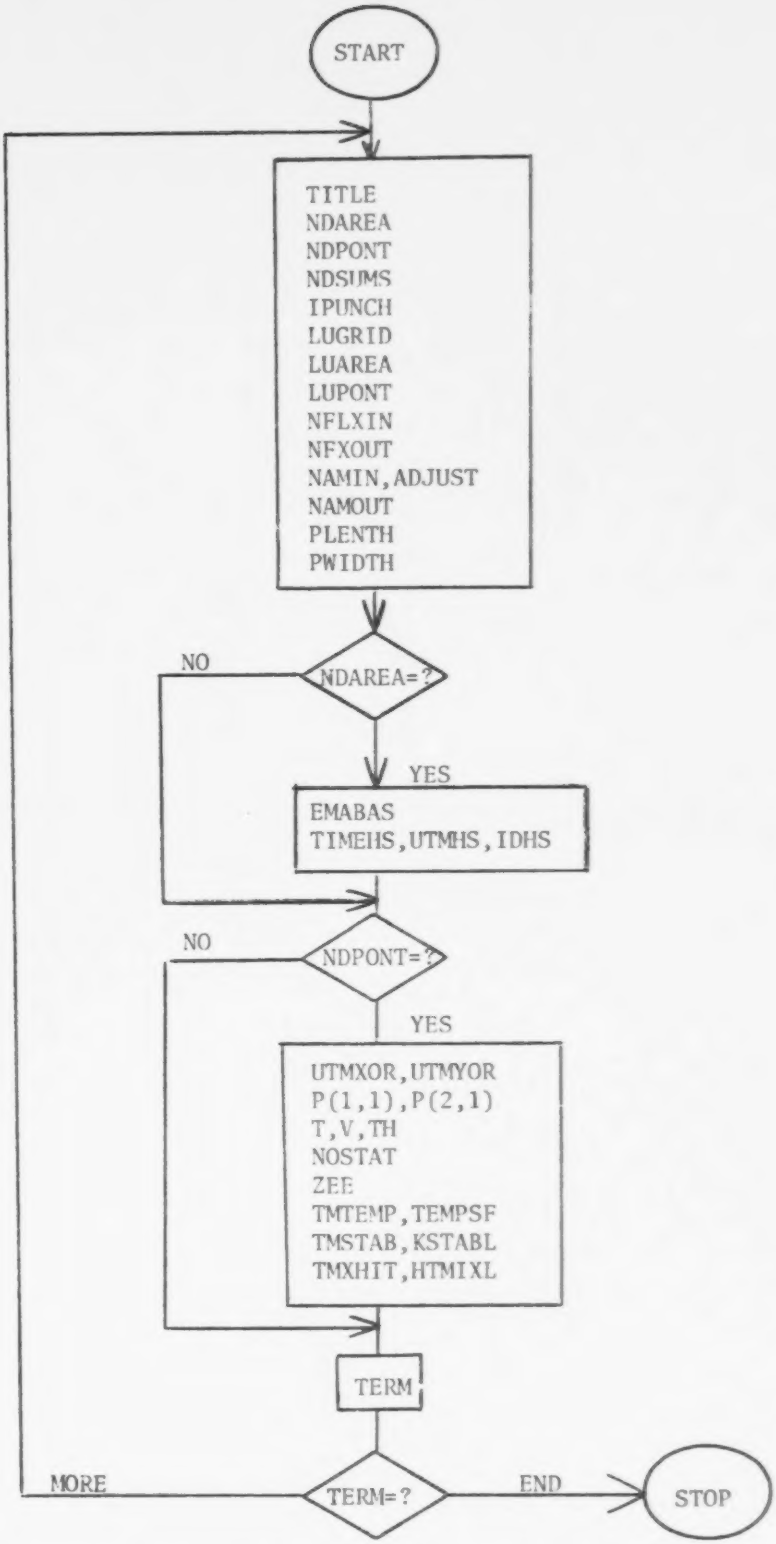


Figure 4-2 EMMOD Input Variable Sequence and Options

4.1.1 General Inputs

The input stream begins with the user-selected page label (TITLE), similar to the first card in the METMOD input deck. This card is followed by four option flags, which are specified by YES or NO inputs. The user first specifies whether (YES) or not (NO) the area source emissions schedule is to be generated in the run. The next option flag similarly indicates whether the point source schedule is desired. The third flag indicates whether or not the user wants the hourly sums of the regional inventories generated and printed. The fourth flag instructs the program to punch the area source and point source emission schedules in a format suitable for input to the chemical-diffusion module. Since the punched output of the program is typically 100 to 500 cards, the use of this option is strongly recommended for error-free data transfer.

Next, the user supplies three cards with the logical unit numbers of the peripheral devices assigned to the three input files (tapes). These integer constants refer to the emissions grid file, the area source file, and point source file, respectively.

The user next specifies the number of species with emissions data in the input files (NFLXIN) and in the output schedules (NFXOUT). The RAPS area source input file has data for eleven species and the point source file has data for ten species. The user should specify NFLXIN equal to eleven in the input deck since the program internally accounts for this discrepancy between the files. The program utilizes data for seven species, thus, the user should set NFXOUT equal to seven in the deck.

Next, the program reads NFLXIN cards, which contain 4-character names for the species in the input files. Adjacent to each species name is a conversion factor to adjust the units of the input data to the units required by the code. The non-unity conversion factors shown in Figure 4-1 are those used to convert the nitrogen oxides (NOX), sulfur oxides (SOX), total hydrocarbons (THC), and carbon monoxide (CO) emissions data from kilograms per hour to moles per hour. A value of unity is specified for the individual hydrocarbon class data since this data is provided in moles per hour. The program does not use the particulate (PART) and dust (DUST) emissions data, so their units are left in kilograms per hour by inputting a conversion factor of one.

The program proceeds to read NFXOUT cards, which contain the 4-character names of the species in the output schedules. These species name cards, as well as those described above, must appear in the order shown in Figure 4-1 for proper labeling of the output data.

The last two general inputs designate the air parcel length and width in kilometers. These are specified to correspond to the minimum grid square size to which the area source data has been allocated. For the St. Louis application, the air parcel is considered to be on kilometer square.

4.1.2 Area Source Emission Control Inputs

The user first specifies a set of NFXOUT cards which indicate the default area source emission rates in moles per hour for the region outside of the emissions grid. The order in which this data is input corresponds to the order of the output species names, as listed in Figure 4-1. For the St. Louis application, the region outside the emissions grid is rural, thus, the default emission rates have been set to zero in the sample problem input stream. There are, of course, cases where non-zero default emission rates would be appropriate. For example, if a user generates a trajectory which parallels a major vehicular artery prior to entering the emissions grid, the user should supply reasonable estimates of the emission rates based on the vehicle-miles-traveled per one-by-one kilometer grid square.

Next, the user supplies a set of cards which describe the trajectory grid square crossing schedule. Each card contains a time in minutes (0-1440 clock), global x and y coordinates of the crossing point, and the corresponding grid square identifier. These cards are punched by the meteorological module to facilitate data transfer. The last card in the set is identified by a negative entry in columns 11-20.

4.1.3 Point Source Emission Control Inputs

The portion of the input deck which applies to the generation of point source emission schedules begins with the specification of global coordinates for the origin of the local coordinate system. This card is followed by data pertaining to the trajectory. The card specifying the trajectory start point in x and y global coordinates is followed by a set of cards with the trajectory node times (minutes), wind velocities (km/hr), and wind directions (polar angles). These cards are output by the meteorological module, and the final entry in the schedule is a negative number in columns 41-50 of the last card.

The user next inputs the number of vertical mesh points and the vertical mesh point elevations in the same format used for the meteorological module input deck. These cards are followed by schedules of surface temperatures, atmospheric stability categories, and mixing height elevations along the trajectory. The times in the schedules are specified on the 0-1440 minute clock, and each of the schedules is terminated by a negative entry in columns 41-50 of the last card. The temperature and stability class data cards are output by the meteorological module. The mixing height elevations are specified externally at hourly intervals in meters above the surface from available meteorological data.

4.2 Description of the RAPS Input Files

The format and contents of the RAPS/St. Louis input files are described in this section to help the user understand the program's present raw data requirements and retrieval software. A common feature of the files is that the data are blocked in groups of 510 binary words per physical record on the tapes. Each file, of course, has a different number of physical records, as described below. The data are read with BUFFER IN statements in the CDC version of the code, but the equivalent binary READ statements for other computers are included as comment cards in the source listings contained in Appendix C.

4.2.1 Emissions Grid File

The emissions grid file contains 32 physical records of data. The first word in the file is the total number of grid squares which is less than or equal to 2000 for the RAPS grid. This word is followed by the specification of eight parameters for each grid square as defined below:

1) Grid Square Identifier

2) Grid Square Area (km^2)

3) UTM Horizontal Coordinate of Southwest Corner (km)

4) UTM Vertical Coordinate of Southwest Corner (m)

5) State Code

6) County Code

7) Average Source Height

8) Surface Roughness Length (cm)

The EMMOD program reads all the data in the file, but stores only the total number of grid squares and two arrays with the grid square identifiers and corresponding areas. The importance of this data stems from the fact that the ordering of the data in the area source emissions file corresponds to that of the grid square identifiers in this file.

4.2.2 Area Source Emissions File

The area source emissions file contains 1056 physical records of emissions data for the single day. The file contains subsets of 44 records for each hour beginning at midnight. Each hourly subset lists the Julian data, number of grid squares, hour (0-23), and emission rates by grid squares for the eleven species listed below:

1) Particulates (kg/hr)

2) Sulfur Oxides (kg/hr weighed as SO_2)

3) Nitrogen Oxides (kg/hr weighed as NO_2)

4) Total Hydrocarbons (kg/hr)

5) Carbon Monoxide (kg/hr)

6) Dust (kg/hr)

7) Nonreactive Hydrocarbons (moles/hr)

8) Alkanes (moles/hr)

9) Alkenes (moles/hr)

10) Aromatics (moles/hr)

11) Aldehydes (moles/hr)

The program is designed to read as much of each hour's data as necessary to satisfy the list of grid squares traversed by the trajectory during a a particular hour. See Section 4.4 for a description of these procedures.

4.2.3 Point Source Emissions File

The point source emissions file may contain up to 408 physical records for one day. The file is structured with hourly subsets of data, and the number of point sources considered may vary from hour to hour. The file contains data for up to 505 point sources, which translates to a maximum of 17 records per hour. Each hourly subset contains the date and hour stored as a 7-character word (YRDAYHR = year, Julian day, hour) and the number of sources for that hour. These two words are followed by the seventeen parameters described below for each point source.

1) UTM Horizontal Coordinate (km)
2) UTM Vertical Coordinate (km)
3) Stack Height (m)
4) Stack Diameter (m)
5) Stack Effluent Temperature (^{o}C)
6) Stack Effluent Volumetric Flow Rate (M^3/min)
7) Particulates (kg/hr)
8) Sulfur Oxides (kg/hr weighed as SO_2)
9) Nitrogen Oxides (kg/hr weighed as NO_2)
10) Total Hydrocarbons (kr/hr)
11) Carbon Monoxide (kg/hr)
12) Nonreactive Hydrocarbons (moles/hr)
13) Alkanes (moles/hr)
14) Alkenes (moles/hr)
15) Aromatics (moles/hr)
16) Aldehydes (moles/hr)
17) Stack Identification Code

The program reads all the data for each hour of the trajectory and stores all the data except the stack diameters, particulates, total hydrocarbons, nonreactive hydrocarbons, and stack identification codes.

4.3 Description of the Outputs

This section describes the printed and punched output obtained from the EMMOD program. The output includes a listing of the input deck produced by PREDAT, verification of portions of the input data, and listings of the area and point source emissions data along the trajectory. The program also prints the hourly sums of the regional emission inventory when the user selects this option.

4.3.1 Area Source Emission Outputs

Following the PREDAT input deck listing shown in Figure 4-1, the program lists verification of the trajectory grid square crossing schedule. Figure 4-3 shows the first page of output for the sample problem schedule, containing the crossing times, global x and y coordinates of the crossing points, and the grid square identifiers. The reader may observe that additional entries have been placed in the schedule for the even hours.

Next, the program lists the hourly regional area source emission sums when this option is selected. Figure 4-4 shows the emission rate sums for the eleven species with input data. These regional emission rates are tabulated in moles per hour with the exception of particulates and dust, which are listed in kilogram per hour. The reader may note the message from subroutine AREAEM shown in the figure, which informs the reader that due to parity errors in the file, the regional sums could not be accurately tabulated for a particular hour. The program may use some data from previous hours when it encounters this problem, as discussed in Section 4.4.

The program then lists several pages of output with area source emission rates corresponding to the grid square crossing schedule. Figure 4-5 shows a sample of the schedule, listing the crossing time, grid square identifier, and emission rates in moles per hour. At the end of this listing, the program provides the total moles of each species entrained into the air parcel during the trajectory (not shown). This

```
ERT EMISSIONS MODULE     --     ST. LOUIS     --     6/29/76   (181)
TRAJECTORY DATA BY GRID SQUARE
              TIME         UTM-X        UTM-Y           ID
              360.00       680.44       4224.05           0
              394.40       683.06       4229.87          64
              420.00       685.19       4234.15          64
              421.46       685.32       4234.39          72
              426.39       685.32       4235.00          65
              480.00       689.52       4242.10          65
              486.07       690.00       4242.90          81
              496.41       692.03       4245.00          82
              511.58       695.00       4248.08          99
              521.06       696.86       4250.00        2028
              537.08       700.00       4253.25         129
              540.00       700.75       4253.62         129
              551.01       703.50       4255.00         130
              556.62       705.00       4255.59        2039
              564.13       707.00       4256.38        2048
              570.04       708.57       4257.00        2049
              571.63       709.00       4257.17        2058
              579.14       711.00       4257.96        2066
              586.65       713.00       4258.75        2076
              589.06       713.64       4259.00        2077
              594.15       715.00       4259.53        2093
              598.58       716.18       4260.00        2094
              600.00       716.52       4260.16        2094
              601.97       717.00       4260.39        2115
```

Figure 4-3

```
ERT EMISSIONS MODULE      --     ST. LOUIS      --     6/29/76   (181)
REGIONAL AREA SOURCE EMISSION SUMS                              (MOLES/HOUR)
 TIME        PART       SOX       NOX       THC        CO       DUST       NRHC       PARF       OLEF       AROM       ALDE
 360.0    1.408E+05 1.734E+04 4.260E+05 6.660E+05 7.681E+06 1.130E+04 7.687E+04 1.536E+05 1.032E+05 4.128E+04 2.835E+04
 420.0    1.304E+05 1.930E+04 4.473E+05 6.589E+05 7.570E+06 1.020E+04 7.665E+04 1.552E+05 1.006E+05 3.941E+04 2.665E+04
 480.0    1.465E+05 1.867E+04 4.073E+05 6.982E+05 7.100E+06 8.481E+03 8.665E+04 1.600E+05 1.037E+05 4.473E+04 3.040E+04
 540.0    1.485E+05 1.895E+04 4.183E+05 7.070E+05 7.378E+06 8.915E+03 8.762E+04 1.596E+05 1.055E+05 4.610E+04 3.136E+04
 600.0    1.362E+05 1.956E+04 4.410E+05 7.472E+05 7.966E+06 9.831E+03 9.217E+04 1.679E+05 1.124E+05 4.895E+04 3.337E+04
 660.0    1.287E+05 2.016E+04 4.642E+05 7.911E+05 8.573E+06 1.078E+04 9.717E+04 1.777E+05 1.199E+05 5.189E+04 3.545E+04
 720.0    1.477E+05 1.981E+04 4.502E+05  .678E+05 8.212E+06 1.021E+04 9.452E+04 1.727E+05 1.158E+05 5.022E+04 3.421E+04
SUBROUTINE AREAEM ENCOUNTERED PARITY ERRORS  -- SOME DATA FROM PREVIOUS HOUR USED
SUBROUTINE AREAEM ENCOUNTERED PARITY ERRORS  -- SOME DATA FROM PREVIOUS HOUR USED
```

Figure 4-4

```
ERT EMISSIONS MODULE      --     ST. LOUIS      --      6/29/76   (181)
AREA SOURCE EMISSION RATES     --     MOLES/HR
   TIME       ID         NOX         PARF        OLEF        AROM        ALDE        CO          SOX
   360.00        0    0.          0.          0.          0.          0.          0.          0.
   394.40       64    1.5358E+00  1.1579E-01  7.5503E-02  5.2732E-02  3.7162E-02  9.7460E+00  8.3542E-02
   420.00       64    1.5384E+00  1.1581E-01  7.5518E-02  5.2737E-02  3.7177E-02  9.7471E+00  8.3542E-02
   421.46       72    2.6094E+00  7.7094E-01  5.1429E-01  2.2089E-01  1.4090E-01  3.9558E+01  1.1636E-01
   426.39       65    3.4124E+00  9.3613E-01  6.9291E-01  3.3134E-01  2.0666E-01  5.7848E+01  1.4653E-01
   480.00       65    3.1214E+00  8.2712E-01  6.0673E-01  2.9523E-01  1.8510E-01  5.0896E+01  1.3716E-01
   486.07       81    2.8224E+00  1.1907E+00  6.6651E-01  2.5071E-01  1.5186E-01  4.114[illegible]E+01  1.2327E-01
   496.41       82    1.7249E+00  5.4565E-01  2.3331E-01  8.3888E-02  4.8603E-02  1.1679E+01  8.8268E-02
   511.58       99    1.2834E+00  1.5005E-01  8.3601E-02  6.9617E-02  3.9611E-02  1.0059E+01  6.8186E-02
   521.06     2028    1.3106E+00  3.1026E-01  1.4598E-01  6.5374E-02  3.9885E-02  1.0229E+01  6.9496E-02
   537.08      129    9.3589E-01  3.0628E-01  1.2538E-01  6.3952E-02  3.0194E-02  7.2528E+00  4.8335E-02
   540.00      129    9.3973E-01  2.8878E-01  1.1898E-01  6.3958E-02  3.0214E-02  7.2544E+00  4.8335E-02
   551.01      130    1.3400E+00  2.4266E+00  1.1248E+00  3.6323E-01  2.3463E-01  2.4126E+01  7.7250E-02
   556.62     2039    2.7680E+00  1.6504E-01  8.8494E-02  7.3531E-02  5.0729E-02  1.0378E+01  1.5134E-01
   564.13     2048    6.0738E-01  1.1611E-01  4.4769E-02  5.4346E-02  2.0984E-02  4.9002E+00  3.0428E-02
   570.04     2049    6.0738E-01  1.1611E-01  4.4769E-02  5.4346E-02  2.0984E-02  4.9002E+00  3.0428E-02
   571.63     2058    1.1478E+00  5.6061E-01  1.4833E-01  2.6501E-01  6.7663E-02  1.1965E+01  3.3576E-02
   579.14     2066    2.4138E+00  2.8468E+00  8.6321E-01  7.0197E-01  1.6614E-01  2.6755E+01  4.6356E-02
   586.65     2076    1.6766E+01  1.4414E+01  8.0023E+00  2.5849E+00  1.5873E+00  4.6777E+02  4.5901E-01
   589.06     2077    3.6274E+00  8.2383E+00  3.1679E+00  6.0526E-01  2.3172E-01  5.3355E+01  9.3886E-02
   594.15     2093    8.0745E+00  8.7550E+00  6.5108E+00  3.8904E+00  2.5968E+00  3.3657E+02  3.1749E-01
   598.58     2094    1.5456E+01  3.0484E+01  1.7883E+01  7.4558E+00  5.1314E+00  6.6376E+02  6.3736E-01
   600.00     2094    1.6795E+01  3.1038E+01  1.8348E+01  7.6546E+00  5.2679E+00  7.0543E+02  6.7479E-01
   601.97     2115    1.6911E+01  1.0559E+01  7.3760E+00  3.4778E+00  2.1121E+00  5.9894E+02  5.4346E-01
```

Figure 4-5

portion of the output is followed by a listing of the same area source emission rate schedule in the code's units of mole fraction-meters per minute. This schedule is illustrated in Figure 4-6, which shows the global coordinates of the grid square crossing points, in addition to the crossing times and emission rates. This schedule concludes the printed output from the area source portion of the program.

4.3.2 Point Source Emission Outputs

The output from the point source portion of the program begins with a verification of the trajectory description (see Figure 4-7). This page of output is followed by verification of the temporal schedules for atmospheric stability class, the surface temperature, and mixing height. Examples of these verification outputs are shown in Figure 4-8.

The program prints the regional sums of the point source emissions for each hour of the trajectory when this option is designated in the input deck. Figure 4-9 is an example of this printout, with the regional emissions rates of all the input species listed in units of moles per hour, except particulates, which remain in kilograms per hour.

The program lists data concerning the point sources with emissions entrained into the air parcel. As shown in Figure 4-10, the program prints the effective time at which a source is passed, the fraction of its emissions which will be entrained into the air parcel moving along the trajectory, the perpendicular distance from the trajectory centerline to the source, and the emission rates for the output species in moles per hour. The treatment of point source emissions in the Lagrangian reference frame is quite complex; therefore, the reader is referred to Section 4.4 for an explanation of the methods used to determine the effective passing time and the fraction of emissions entrained into the air parcel.

Next, the program prints the actual number of moles entrained into the air parcel from each point source and its vertical distribution. Figure 4-11 is an example of this listing, which first shows the output species names and the ranges of elevations of each vertical cell to which the emissions are allocated. These ranges of elevations are determined from the vertical mesh geometry input to the program. For each point source in the schedule, the listing shows the effective

```
ERT EMISSIONS MODULE     --     ST. LOUIS     --     6/29/76   (181)
AREA SOURCE EMISSION FLUXES                         MOLE FRACTION - METERS/MINUTE
   TIME      UTM-X      UTM-Y     NOX          PARF         OLEF         AROM         ALDE         CO           SOX
   360.0     680.4     4224.0  0.           0.           0.           0.           0.           0.           0.
   394.4     683.0     4229.9  6.2948E-10   4.7461E-11   3.0947E-11   2.1613E-11   1.5232E-11   3.9946E-09   3.4242E-11
   420.0     685.2     4234.1  6.3056E-10   4.7468E-11   3.0953E-11   2.1616E-11   1.5238E-11   3.9951E-09   3.4242E-11
   421.5     685.3     4234.4  1.0695E-09   3.1599E-10   2.1080E-10   9.0539E-11   5.7753E-11   1.6214E-08   4.7693E-11
   426.4     685.3     4235.0  1.3987E-09   3.8370E-10   2.8401E-10   1.3581E-10   8.4703E-11   2.3710E-08   6.0059E-11
   460.0     689.5     4242.1  1.2794E-09   3.3902E-10   2.4868E-10   1.2101E-10   7.5869E-11   2.0861E-08   5.6217E-11
   486.1     690.0     4242.9  1.1568E-09   4.8803E-10   2.7318E-10   1.0276E-10   6.2245E-11   1.6862E-08   5.0526E-11
   496.4     692.0     4245.0  7.0699E-10   2.2365E-10   9.5628E-11   3.4384E-11   1.9921E-11   4.7868E-09   3.6179E-11
   511.6     695.0     4248.1  5.2604E-10   6.1504E-11   3.4266E-11   2.8534E-11   1.6236E-11   4.1231E-09   2.7948E-11
   521.1     696.9     4250.0  5.3719E-10   1.2717E-10   5.9833E-11   2.6795E-11   1.6348E-11   4.1926E-09   2.8485E-11
   537.1     700.0     4253.2  3.8360E-10   1.2554E-10   5.1388E-11   2.6212E-11   1.2376E-11   2.9728E-09   1.9811E-11
   540.0     700.7     4253.6  3.8517E-10   1.1836E-10   4.8767E-11   2.6215E-11   1.2384E-11   2.9734E-09   1.9611E-11
   551.0     703.5     4255.0  5.4923E-10   9.9462E-10   4.6101E-10   1.4888E-10   9.6169E-11   9.8887E-09   3.1663E-11
   556.6     705.0     4255.6  1.1345E-09   6.7646E-11   3.6272E-11   3.0139E-11   2.0793E-11   4.2536E-09   6.2029E-11
   564.1     707.0     4256.4  2.4895E-10   4.7590E-11   1.8350E-11   2.2275E-11   8.6010E-12   2.0085E-09   1.2472E-11
   570.0     708.6     4257.0  2.4895E-10   4.7590E-11   1.8350E-11   2.2275E-11   8.6010E-12   2.0085E-09   1.2472E-11
   571.6     709.0     4257.2  4.7047E-10   2.2978E-10   6.0799E-11   1.0862E-10   2.7733E-11   4.9042E-09   1.3762E-11
   579.1     711.0     4258.0  9.8934E-10   1.1668E-09   3.5381E-10   2.8772E-10   6.8099E-11   1.0966E-08   1.9000E-11
   586.7     713.0     4258.7  6.8719E-09   5.9080E-09   3.2800E-09   1.0595E-09   6.5059E-10   1.9173E-07   1.8814E-10
   589.1     713.6     4259.0  1.4868E-09   3.3767E-09   1.2984E-09   2.4808E-10   9.4978E-11   2.1869E-08   3.8482E-11
   594.2     715.0     4259.5  3.3096E-09   3.5885E-09   2.6686E-09   1.5946E-09   1.0644E-09   1.3795E-07   1.3013E-10
   598.6     716.2     4260.0  6.3348E-09   1.2495E-08   7.3298E-09   3.0559E-09   2.1032E-09   2.7206E-07   2.6124E-10
   600.0     716.5     4260.2  6.8837E-09   1.2722E-08   7.5204E-09   3.1374E-09   2.1592E-09   2.8914E-07   2.7658E-10
   602.0     717.0     4260.4  6.9314E-09   4.3280E-09   3.0233E-09   1.4255E-09   8.6568E-10   2.4549E-07   2.2275E-10
```

Figure 4-6

```
ERT EMISSIONS MODULE     --     ST. LOUIS     --     6/29/76   (181)

     TRAJECTORY DATA

LEG       TIME        X          Y          VELOCITY      THETA
NO.       (MIN)      (KM)       (KM)        (KM/HR)     (DEGREES)

 1        360.0     680.44     4224.05      11.090       66.270

 2        420.0     684.90     4234.20       8.421       62.380

 3        480.0     688.81     4241.66      16.930       45.980

 4        540.0     700.57     4253.84      17.180       21.530

 5        600.0     716.55     4260.14      15.400       29.040

 6        660.0     730.02     4267.62      15.380       25.000

 7        720.0     743.96     4274.12      21.630       42.260

 8        780.0     759.96     4288.66      18.180       46.720

 9        840.0     772.43     4301.90      19.680       30.470

10        900.0     789.39     4311.88      21.530       28.280

11        960.0     808.35     4322.08      24.820       15.540

12       1020.0     832.26     4328.73      24.820       15.540

13       1080.0     856.18     4335.38      24.820       15.540
```

Figure 4-7

STABILITY CLASS DATA

TIME	CLASS
360	2
660	5
720	2
900	3
960	2

SURFACE TEMPERATURE DATA

TIME	TEMPERATURE
360	21.5
420	22.4
480	24.4
540	26.5
600	27.7
660	26.5
720	28.5
780	29.6
840	29.4
900	29.3
960	29.7
1020	29.6

MIXING LAYER DEPTH DATA

TIME	DEPTH
360	200.0
420	270.0
480	340.0
540	410.0
600	490.0
660	600.0
720	710.0
780	820.0
840	930.0
900	1040.0
960	1080.0

Figure 4-8

```
ERT EMISSIONS MODULE     --     ST. LOUIS     --     6/29/76    (181)

REGIONAL POINT SOURCE EMISSION SUMS                          (MOLES/HOUR)

 TIME       PART        SOX        NOX        THC         CO        NRHC       PARF       OLEF       AROM       ALDE

 360.0   5.978E+03  1.682E+06  7.693E+05  1.148E+05  6.114E+05  2.509E+04  2.313E+04  4.987E+03  1.420E+04  4.113E+02

 420.0   7.614E+03  2.083E+06  8.422E+05  1.780E+05  6.205E+05  3.078E+04  3.763E+04  5.319E+03  2.321E+04  4.528E+02

 480.0   7.674E+03  2.121E+06  6.566E+05  1.781E+05  6.310E+05  3.102E+04  3.763E+04  5.327E+03  2.321E+04  4.609E+02

 540.0   7.776E+03  2.201E+06  9.079E+05  1.785E+05  6.278E+05  3.158E+04  3.765E+04  5.346E+03  2.322E+04  4.798E+02

 600.0   7.863E+03  2.214E+06  9.193E+05  1.787E+05  6.285E+05  3.184E+04  3.766E+04  5.354E+03  2.322E+04  4.883E+02

 660.0   7.857E+03  2.138E+06  8.712E+05  1.785E+05  6.174E+05  3.147E+04  3.765E+04  5.342E+03  2.322E+04  4.760E+02

 720.0   7.864E+03  2.142E+06  8.811E+05  1.786E+05  6.179E+05  3.175E+04  3.766E+04  5.351E+03  2.322E+04  4.851E+02

 780.0   7.858E+03  2.131E+06  8.663E+05  1.783E+05  6.172E+05  3.132E+04  3.764E+04  5.337E+03  2.322E+04  4.709E+02

 840.0   7.763E+03  2.128E+06  8.641E+05  1.783E+05  6.170E+05  3.125E+04  3.764E+04  5.335E+03  2.322E+04  4.687E+02

 900.0   4.315E+03  2.129E+06  8.635E+05  1.526E+05  6.169E+05  2.925E+04  3.161E+04  5.213E+03  1.937E+04  4.673E+02

 960.0   4.268E+03  2.124E+06  8.611E+05  1.519E+05  6.167E+05  2.915E+04  3.141E+04  5.210E+03  1.933E+04  4.646E+02

1020.0   4.185E+03  2.120E+06  8.578E+05  1.518E+05  6.164E+05  2.905E+04  3.140E+04  5.207E+03  1.933E+04  4.616E+02
```

Figure 4-9

```
ERT EMISSIONS MODULE     --     ST. LOUIS     --     6/29/76   (181)
POINT SOURCE EMISSION RATES   --   MOLES/HOUR
    TIME      FRACTION   DISTANCE    NOX          PARF         OLEF         AROM         ALDE         CO           SOX
    709.80    .1953       .206       0.           0.           0.           0.           0.           0.           0.
    711.40    .1929      1.202       0.           7.4593E+02   4.3073E+03   0.           1.5240E+00   0.           0.
    711.48    .1896       .347       0.           0.           0.           0.           0.           7.4596E+02   4.3073E+03
    711.50    .1898       .334       0.           1.5240E+00   0.           0.           0.           0.           8.1746E+00
    719.64    .1725       .617       8.1746E+00   0.           0.           0.           0.           0.           0.
    721.87    .1662       .961       2.0038E-01   6.6793E-02   2.0038E-01   0.           7.4610E+02   4.3073E+03   0.
    721.87    .1662       .961       1.5240E+00   0.           0.           8.5893E-03   3.2580E-02   8.1746E+00   0.
    722.70    .1677       .655       0.           5.4431E+00   0.           0.           0.           0.           0.
    722.70    .1677       .655       0.           0.           0.           0.           7.5530E+02   4.3026E+03   0.
    725.51    .1541      1.203       0.           0.           7.4598E+02   4.3073E+03   0.           1.5240E+00   0.
    728.18    .1757       .313       0.           2.9484E+01   0.           0.           0.           0.           0.
    728.42    .1668       .764       0.           0.           0.           7.5473E+02   4.3023E+03   0.           1.9202E+00
    728.42    .1668       .764       0.           0.           0.           3.3566E+01   0.           0.           0.
    728.42    .1668       .764       0.           0.           0.           0.           0.           0.           7.5469E+02
    728.69    .1402      1.590       0.           0.           0.           8.1746E+00   0.           0.           0.
    728.95    .1760       .307       0.           0.           0.           0.           7.5444E+02   4.3027E+03   0.
    728.97    .1761       .300       0.           3.7467E+02   0.           0.           0.           0.           0.
    729.04    .1776       .093       0.           0.           0.           0.           0.           0.           0.
    729.29    .1758       .337       0.           7.5485E+02   4.3027E+03   0.           2.8956E+00   0.           0.
    729.30    .1691       .688       0.           0.           0.           1.7690E+01   0.           0.           0.
    729.30    .1691       .688       0.           0.           0.           0.           0.           0.           7.5499E+02
    729.30    .1691       .688       0.           0.           0.           0.           0.           7.5499E+02   4.3023E+03
    729.30    .1691       .688       0.           2.3774E+00   0.           0.           0.           5.5792E+01   0.
    729.30    .1691       .688       0.           0.           0.           0.           0.           0.           0.
```

Figure 4-10

```
EKT EMISSIONS MODULE     --     ST. LOUIS     --     6/29/76    (181)

POINT SOURCE EMISSIONS   --   MOLES AND VERTICAL DISTRIBUTION

     SPECIES              NOX        PARF        OLEF        AROM        ALDE          CO          SOX
     ELEVATIONS          0-  60     60- 210     210- 450    450- 900    900-1200
  TIME

  709.80              0.          3.7508E+00  1.3748E+00  0.          0.           0.           0.
                       .1283       .2913       .3656       .2149      0.0000

  711.40              3.1277E+01  0.          0.          0.          0.           0.           8.0815E+01
                       .1787       .2098       .4552       .1564      0.0000

  711.48              1.7948E+00  1.3127E-01  2.8174E-01  1.8576E-02  5.6348E-02  5.1601E+01   1.6125E+00
                       .1389       .2223       .4210       .2178      0.0000

  711.50              1.7963E+00  1.3138E-01  2.8197E-01  1.8592E-02  5.6395E-02  5.1643E+01   1.6139E+00
                       .1389       .2223       .4210       .2178      0.0000

  719.64              0.          2.1834E+00  8.0031E-01  0.          0.           0.           0.
                       .1325       .3020       .3877       .1778      0.0000

  721.87              0.          6.0416E+00  0.          8.4123E-01  0.           0.           0.
                       .1416       .3200       .4085       .1299      0.0000

  721.87              0.          9.1424E-04  0.          2.1213E-04  0.           0.           0.
                       .1416       .3200       .4085       .1299      0.0000

  722.70              0.          6.2444E+00  2.2888E+00  0.          0.           0.           0.
                       .1562       .3083       .3409       .1946      0.0000

  722.70              0.          1.1765E+00  4.3123E-01  0.          0.           0.           0.
                       .1533       .3086       .3421       .1960      0.0000

  725.51              0.          3.3133E+00  1.2144E+00  0.          0.           0.           0.
                       .1329       .3013       .3873       .1785      0.0000

  728.18              0.          0.          0.          0.          0.           0.           2.6075E+01
                       .1599       .2947       .3210       .2244      0.0000

  728.42              2.7704E-02  3.8232E-05  3.8232E-05  1.2744E-05  3.8232E-05  6.4478E-03   9.9561E-05
                       .1615       .3133       .3445       .1808      0.0000

  728.42              2.7704E-02  3.8232E-05  3.8232E-05  1.2744E-05  3.8232E-05  6.4478E-03   9.9561E-05
                       .1615       .3133       .3445       .1808      0.0000

  728.42              2.7704E-02  3.8232E-05  3.8232E-05  1.2744E-05  3.8232E-05  6.4478E-03   9.9561E-05
                       .1615       .3133       .3445       .1808      0.0000

  728.69              0.          4.5731E-01  1.6762E-01  0.          0.           0.           0.
                       .1338       .3034       .3876       .1752      0.0000

  728.95              0.          0.          0.          0.          0.           0.           1.4749E+00
                       .1590       .3090       .3393       .1927      0.0000

  728.97              0.          0.          0.          0.          0.           0.           1.0145E+00
                       .1673       .3082       .3356       .1889      0.0000
```

Figure 4-11

passing time and moles of each species entrained on one line. The following line indicates the normalized vertical distribution of the emissions with respect to the vertical cells. At the conclusion of this listing (not shown), the program prints the total number of moles entrained into the air parcel during the trajectory.

Lastly, the program prints the same schedule of point source emissions in the units required by the chemical-diffusion module. Figure 4-12 shows an example of this listing where the species emissions are shown in the units of mole-fraction meters.

4.3.3 Punched Outputs

The program punches, on logical unit 1, the area source emissions schedule shown in Figure 4-6, in a format suitable for input to the chemical-diffusion module. Similarly, the program punches out the point source emissions data shown in Figure 4-12 and the vertical distributions shown in Figure 4-11 for each source in the schedule.

4.4 Computational Procedures

The computational procedures of the EMMOD program will be discussed in the order in which they occur in the program. The computational procedures of the program are quire simple for area source emissions and quite complex for point source emissions. The complexity in the treatment of point sources stems from the fact that the approximate effects of lateral (cross-trajectory) dispersion of point source emissions are accounted for in the emission schedules.

4.4.1 Area Source Emission Tabulations

Once the program has read the general inputs, the grid square crossing schedule, and default emissions from the card deck, the program reads and stores the emission grid square identifiers and areas (subroutine GRIDIT). It then proceeds to insert entries for the even hours in the grid square crossing schedule (subroutine ADHOUR). These additional entries are necessary since new emissions data is available for each hour.

Program control is then passed to subroutine AREAEM, which reads the area source emissions file and generates the schedule of emissions.

```
ERT EMISSIONS MODULE     --     ST. LOUIS     --     6/29/76   (181)
POINT SOURCE EMISSIONS  --  MOLE FRACTION-METERS
  TIME                    NOX         PARF        OLEF        AROM        ALDE        CO          SOX
  709.80                0.          9.2240E-08  3.3810E-08  0.          0.          0.          0.
  711.40                7.6917E-07  0.          0.          0.          0.          0.          1.9874E-06
  711.48                4.4139E-08  3.2283E-09  6.9287E-09  4.5684E-10  1.3857E-09  1.2690E-06  3.9656E-08
  711.50                4.4175E-08  3.2310E-09  6.9344E-09  4.5722E-10  1.3869E-09  1.2700E-06  3.9689E-08
  719.64                0.          5.3696E-08  1.9682E-08  0.          0.          0.          0.
  721.87                0.          1.4858E-07  0.          2.0688E-08  0.          0.          0.
  721.87                0.          2.2484E-11  0.          5.2169E-12  0.          0.          0.
  722.70                0.          1.5356E-07  5.6287E-08  0.          0.          0.          0.
  722.70                0.          2.8933E-08  1.0605E-08  0.          0.          0.          0.
  725.51                0.          8.1482E-08  2.9866E-08  0.          0.          0.          0.
  728.18                0.          0.          0.          0.          0.          0.          6.4126E-07
  728.42                6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13  9.4022E-13  1.5857E-10  2.4485E-12
  728.42                6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13  9.4022E-13  1.5857E-10  2.4485E-12
  728.42                6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13  9.4022E-13  1.5857E-10  2.4485E-12
  728.69                0.          1.1246E-08  4.1222E-09  0.          0.          0.          0.
  728.95                0.          0.          0.          0.          0.          0.          3.6272E-08
  728.97                0.          0.          0.          0.          0.          0.          2.4949E-08
  729.04                0.          0.          0.          0.          0.          0.          1.0422E-06
  729.29                0.          0.          0.          0.          0.          0.          2.4811E-07
  729.30                0.          0.          0.          3.6492E-09  0.          0.          0.
  729.30                0.          0.          0.          1.8246E-08  0.          0.          0.
  729.30                0.          0.          0.          9.2889E-09  0.          0.          0.
  729.30                0.          0.          0.          3.9312E-08  0.          0.          0.
  729.30                0.          0.          0.          3.5663E-08  0.          0.          0.
```

Figure 4-12

AREAEM skips data in the file pertaining to hours before the trajectory start time. It then enters an hourly time cycle to determine the largest grid square identifier in the current hour's grid square crossing list. It proceeds to read the data in the file up to the point where it passes the data for the largest identifier, and skips the remaining data for that hour. If the user has requested printing of the regional sums, of course, all of the data for that hour are read. The program then locates the species' emission rates for the grid squares being crossed, and adjusts the rates to represent moles per hour for grid squares with areas defined by the air parcel length and width inputs. The hourly time cycle is repeated until the program has located and adjusted the emission rates for all grid squares crossed by the trajectory.

If the program encounters a parity error when reading the area source file, it uses the data from the previous hour for the subset of grid squares with unreadable data.

The program prints the schedule of area source emission rates in moles per hour, and then proceeds to convert the data to the units required by the chemical-diffusion module. The required conversion of moles per hour to mole fraction-meters per minute is described below.

Let: ϕ = emission rate in moles per hour

M = molecular weight of air (28.97)

ρ = density of air (grams per cubic meters)

A = air parcel area (square kilometers)

α = 10^6 square meters per square kilometer

β = 60 minutes per hour

ϕ' = emission rate in mole fraction-meters per minute

Then:

$$\phi' = \frac{M\phi}{\alpha\beta\rho A} \qquad (4\text{-}1)$$

Following this conversion, the program proceeds to print and, optionally, punch the area souce emissions schedule.

4.4.2 Point Source Emission Tabulations

The objective of the point source algorithm is to determine the pollutant emissions entrained into a Lagrangian air parcel travelling across the modeling region. To accomplish this objective, the program examines the meteorologically-dependent characteristics of point source plumes intersecting the trajectory path. Plumes are treated on an individual basis, assuming that the lateral and vertical dispersion of emissions is Gaussian in nature. Dispersion of effluent from tall stacks and short (less than 100m) stacks are simulated by using separate sets of diffusion coefficients. The rise of an individual plume near the trajectory is calculated with consideration of inversion penetration by buoyant plumes.

To comprehend the algorithm, the reader should first understand how plumes are believed to be entrained into a fixed size Lagrangian air parcel. Figure 4-13 schematically illustrates a side view of a parcel passing a point source. Assuming for the moment that the source is within the air parcel, the parcel entrains the emissions for the time period required to pass the source. Once these emissions have been entrained, lateral dispersion occurs, which may carry a portion of the pollutants out the sides of the parcel. Similarly, the parcel may pass a source whose perpendicular distance from the trajectory's center line is greater than the parcel's half width, and its plume may disperse into the parcel at a distance downwind of the source. This lateral spreading of plumes into and out of air parcels is illustrated in Figure 4-14.

According to the K-theory of diffusion, lateral dispersion of entrained emissions continues until the lateral concentration gradients become negligible. For practical considerations, the program makes the assumption that the majority of lateral spreading occurs during a fixed time interval, currently set at one hour. Hence, the portion of stack's emissions that remains in the air parcel after one hour of dispersion forms the basis of the point source emissions schedule used by the chemical-diffusion module. The vertical distribution of the entrained emissions depends on the meteorological conditions and individual stack parameters; and, these distributions are also generated for use by the chemical-diffusion module by the program.

The point source algorithm's computations begin after the trajectory data, vertical mesh, temperatures, stability class, and mixing heights have been read by the main program, and program control is passed to subroutine PONTEM. PONTEM performs several initialization tasks, which include calling SEGMET. SEGMET determines the distances

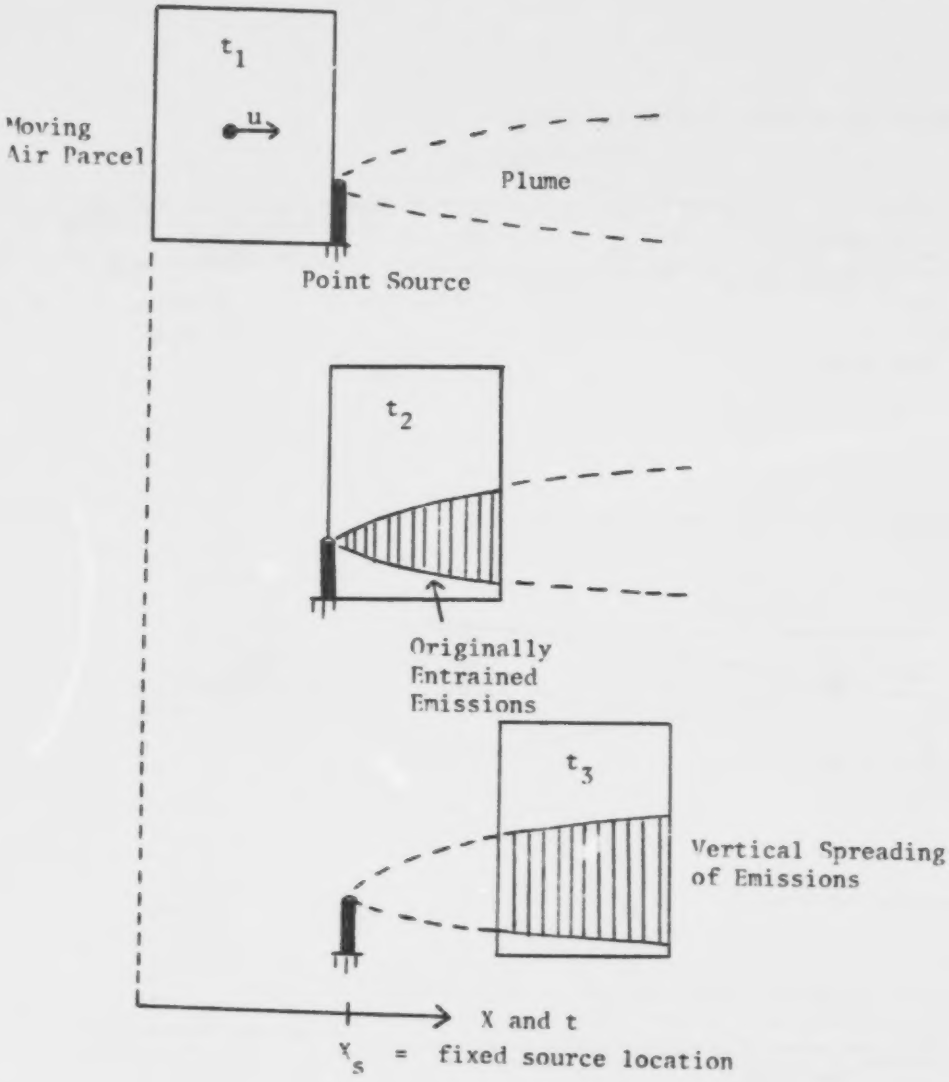
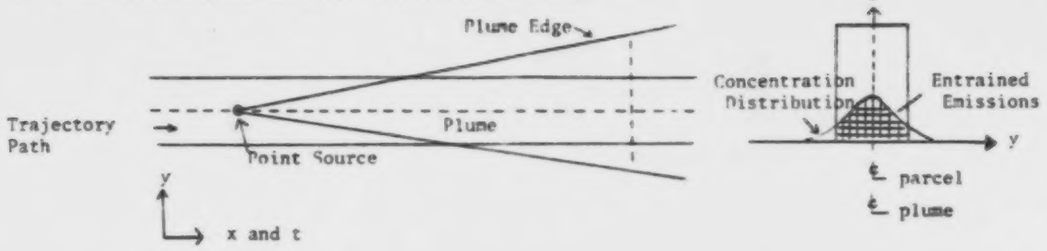
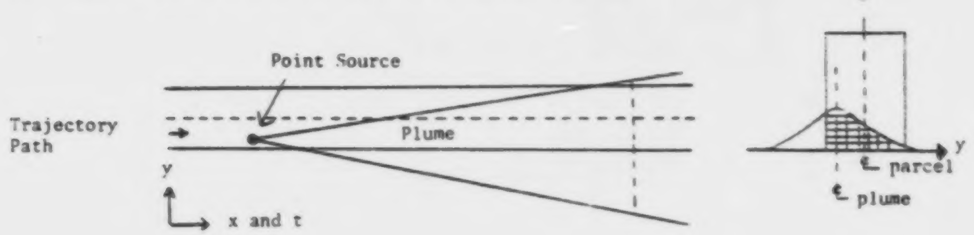

t_1
Moving Air Parcel
u
Plume
Point Source
t_2
Originally Entrained Emissions
t_3
Vertical Spreading of Emissions
X and t
X_s = fixed source location

Figure 4-13 Plume Entrainment in a Lagrangian Air Parcel (Side Views at time = t_1, t_2, t_3)

a) Plume Spreading Out of Parcel (Centerline Passing)



b) Plume Spreading Out of Parcel (Off Centerline Passing)



c) Plume Spreading Into Parcel

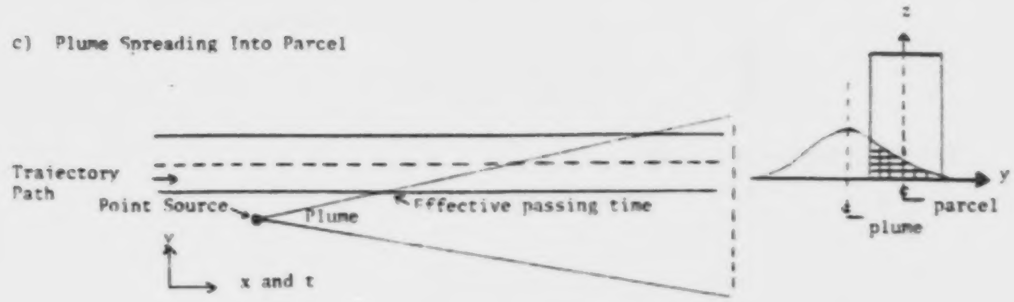


Figure 4-14 Lateral Spreading of Plumes Into and Out of Moving Air Parcels

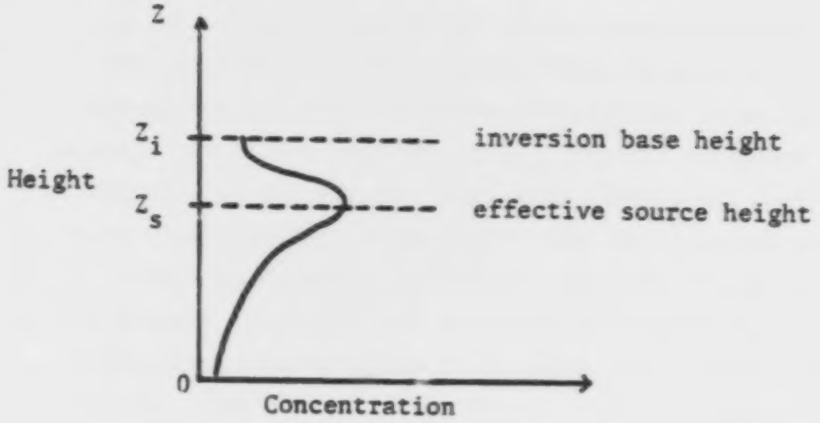


Figure 4-15 Vertical Distribution of Emission Below Inversion Base

between trajectory nodes and the bisector angles between trajectory segments. It also determines certain maximum distance criteria, which are functions of stability class and the maximum segment length. These criteria are subsequently used by subroutine LOCATE to efficiently select a subset of plumes along a segment, which may intersect the trajectory path. PONTEM then calls subroutine STACKS to read the RAPS/St. Louis point source file for the first hour of the trajectory.

Following the initializations tasks, PONTEM passes control to an outer loop, which cycles through each segment of the trajectory. Subroutine GESTAB is called to retrieve the current hour's and subsequent hour's atmospheric stability classes. Also, the time period for entrainment is calculated from the current wind speed and the parcel's length.

Control is then passed to an inner loop, which considers each point source in the current hour's emissions inventory. Subroutine LOCATE examines the location of a source relative to the trajectory segment end points and bisector angles to determine if it meets certain preliminary or relative position criteria. If a source meets the criteria, subroutine PLUMAS is called. Here, special treatment is given to sources passed with a perpendicular distance greater than the parcel half width. PLUMAS determines whether or not the edge of such a plume intersects the side of the parcel within one hour after passing the source. For practical purposes, the program considers the edge of the plume to be the 10% centerline concentration boundary (i.e., $\sigma_y = 2.14$). Plumes which fail this test are ignored. If the plume edge is found to intersect the parcel within the current hour, the program resets the passing time to the time at which the plume edge is intercepted. For sources within the air parcel path, the passing time is simply the time at which parcel center is perpendicular to the source. PLUMAS next evaluates the fraction of entrained emissions which exist in the air parcel one hour after the effective passing time. This computation involves the lateral integration of the Gaussian concentration distribution over the width of the air parcel. Changes in wind speed and/or stability class during the hour are accounted for in the assumed Gaussian distribution.

Given the fraction of emissions entrained from a source, PONTEM proceeds to calculate the vertical characteristics of the plume. Subroutine DHPLUM is called to determine the source's plume rise and inversion penetration fraction, using formulas developed by Briggs, 1969 and

1975. This information requires the following information:

- Ambient and stack effluent temperatures
- Wind speed and atmospheric stability class
- Stack height and volumetric flow rate
- Inversion height or mixing layer depth
- Potential temperature gradient in the inversion layer

The effective plume centerline height is determined from the source height and plume rise. Next, subroutine PARTIT is called to evaluate and integrate the vertical concentration distribution. This distribution is currently evaluated 20 minutes after the time of passing. For a source below the inversion base, the program assumes the distribution is Gaussian with reflections at the surface and inversion base as shown in Figure 4-15. Subroutine PARTIT integrates this profile over each vertical cell to determine the vertical distribution of emissions. The fraction of emissions which penetrate the inversion base are allocated to the cell which bounds the inversion, unless the inversion base is within ten meters of the next vertical cell's lower boundary. When this occurs, the penetration fraction is allocated to the next vertical cell.

The computations within the inner loop are complete when the program has stored the passing time, source emission rates, entrained mass, and vertical distribution for each qualifying source on a trajectory segment. Program control is then returned to the outer loop which updates the temperature and mixing height data and reads another hour's data from the RAP point source file. This cycle of computations is repeated until all the plumes intersected along the trajectory have been evaluated.

PONTEM then orders the data chronologically and converts the units of the entrained emissions to the units required by the chemical-diffusion module. Given the entrained emissions in moles, the conversion is performed as follows:

Let: ϕ = moles entrained

M = molecular weight of air (28.97)

ρ = density of air (grams per cubic meter)

A = air parcel area (square kilometers)

α = 10^6 square meters per square kilometers

ϕ' = mole fraction-meters entrained

Then: $\phi' = \frac{M\phi}{\alpha\rho A}$

Following the units conversion, PONTEM proceeds to print, and optionally punch, the point source emissions schedule.

4.5 Internally Specified Data

The emissions module has numerous internally specified parameters of which the user should be aware. Table 4-2 lists the parameter's code variable names, functions, and location in the code. Only one of the parameters, the density of air, is specific to a particular modeling region. The user should change the surface air density value when applying the model to a region with surface elevations significantly above sea level.

Two additional tables are provided which contain information on plume dispersion coefficients used in the model. Table 4-3 list the plume dispersion coefficients, σ_y and σ_z, for non-tall stacks. Table 4-4 lists the coefficients, a, b, c, and d, used in the formulas to determine tall stack plume dispersion coefficients. The data in the tables was obtained from Turner, 1969, and the ASME, 1968.

4.6 Program Modifications

As previously mentioned, the program has been equipped with retrieval software to read the RAPS grid and emissions files. For adaptation of EMMOD to a region other than St. Louis where the data formats of available data are quite similar in structure, the user may redesign the retrieval software in subroutines GRIDIT, AREAEM, and STACKS to suit the data base.

The point source portion of the program is considered to be quite general. If the user designs a new subroutine STACKS to fill the appropriate arrays with the proper data, the program should run without difficulty. The user should note that the arrays in STACKS are presently dimensioned for 505 point sources per hour, and can easily be enlarged.

The area source portion of the program is not considered to be general enough to utilize without major modifications for most regions other than St. Louis. Users adapting the program to other regions may find it more convenient to acquire ERT's FLXGEN program. The FLXGEN program is a general-purpose regional emissions processor capable of generating area source emission schedules for the chemical-diffusion module. It assembles inventories for up to fifty different area source categories and uses constant-size grid square I and J indices instead of variable-size grid square identifiers.

TABLE 4-2

EMISSIONS MODULE INTERNALLY SPECIFIED DATA

Variable Name	Function	Location
ROAIR	Density of air (gm/meters^3) (currently set equal to 1178)	EMMOD
SIGEDG	The lateral edge of a Gaussian plume, defined by the number of standard deviations from the plume's centerline. (currently equals 2.14)	EMMOD
DTFREZ	The time interval for consideration of lateral diffusion from a point source (currently set at 60 minutes).	EMMOD
VDXFRC	The non-dimensional fraction of DTFREZ at which a plume's vertical distribution is evaluated (currently equals .33).	PONTEM
DTDZ	Potential temperature gradient (^{o}C/m) in the inversion layer (currently equals .0137).	PONTEM
NDY	The number of distances downwind for which lateral dispersion coefficients (σ_y) are tabulated (currently equals 4).	BLOCK DATA
XDY	The distances downwind corresponding to the tabulated lateral dispersion coefficients (km).	BLOCK DATA
SIGYD	The tabulated lateral dispersion coefficients (σ_y) as functions of atmospheric stability class and downwind distances (km).	BLOCK DATA
NDZ	The number of distances downwind for which vertical dispersion coefficients (σ_z) are tabulated (max = 12).	BLCOK DATA
XDZ	The distances downwind corresponding to the tabulated vertical dispersion coefficients (meters).	BLOCK DATA
SIGZD	The tabulated vertical dispersion coefficients (σ_z) as functions of atmospheric stability class and downwind distance (meters).	BLOCK DATA
NCAT	Number of tabulated atmospheric stability classes (max = 6).	BLOCK DATA
ATALL, BTALL	The "a" and "b" coefficients by stability class in the tall stack lateral dispersion coefficient formula $\sigma_y = ax^b$.	BLOCK DATA

TABLE 4-2 (CONTINUED)

Variable Name	Function	Location
CTALL, DTALL	The "c" and "d" coefficients by stability class in the tall stack vertical dispersion coefficient formula $\sigma_z = cx^d$.	BLOCK DATA

TABLE 4-3

PLUME DISPERSION COEFFICIENTS FOR NON-TALL STACKS

a) Lateral Dispersion Coefficients (σ_y in kilometers)

Distance Downwind (km)	Stability Class A	B	C	D	E	F
0.1	.027	.019	.0125	.008	.006	.004
1.0	.212	.157	.104	.068	.050	.034
10.0	1.57	1.19	.84	.555	.410	.273
80.0	9.0	6.8	5.05	3.35	2.73	1.68

b) Vertical Dispersion Coefficients (σ_z in meters)

Distance Downwind (km)	Stability Class A	B	C	D	E	F
0.1	14.	10.8	7.4	4.6	3.5	2.3
0.2	29.5	20.3	13.9	8.5	6.38	4.05
0.3	48.	30.2	20.1	12.1	8.8	5.6
0.5	105.	51.	32.	18.6	13.0	8.5
1.0	450.	110.	61.	31.5	21.3	14.0
2.0	1950.	232.	116.	50.0	33.7	21.5
3.0	4600.	365.	169.	64.5	42.7	26.5
5.0	13562.	640.	267.	89.0	56.0	34.0
10.0	58818.	1350.	500.	137.	79.0	46.5
20.0	255084.	2900.	950.	202.	110.	60.0
50.0	1774117.	7968.	2170.	328.	153.	79.0
100.0	7694071.	17117.	4000.	455.	183.	93.0

TABLE 4-4

PLUME DISPERSION COEFFICIENTS FOR TALL STACKS

Where: $\sigma_y = ax^b$

$\sigma_z = cx^d$

x = downwind distance

Coefficient	A	B	C & D	E & F
a	.40	.36	.32	.31
b	.91	.86	.78	.71
c	.40	.33	.22	.06
d	.91	.86	.78	.71

5. THE CHEMICAL-DIFFUSION MODULE

This section provides instructions for the use of the chemical-diffusion module program KEMOD. The basic function of the program is solution of the atmospheric diffusion equation in the Lagrangian reference frame for the pollutant concentrations at vertically discretized heights. The solution is obtained by numerically integrating finite difference approximations to the equation shown below:

$$\frac{\partial C_i(z,t)}{\partial t} = \frac{\partial}{\partial z}\left(K_z(z,t)\,\frac{\partial C_i(z,t)}{\partial z}\right) + \dot{S}_i + \dot{R}_i(C_i, C_j, \ldots, C_r) \qquad (5\text{-}1)$$

Where: $C_i(z,t)$ = concentration of i^{th} species

$K_z(z,t)$ = eddy diffusivity coefficient

$\dot{S}_i$ = source emissions of i^{th} species

$\dot{R}_i$ = chemical rate of change of i^{th} species

z = elevation above ground

t = time

The program utilizes the schedules of emissions data, K_z coefficients, temperatures, and sky clearness parameters. In addition, the user provides initial concentrations and chemical rate constants for the model. The program solves the equations and outputs the concentrations as functions of time and height along the trajectory.

5.1 Control Parameters and Input Data

The input parameters will be discussed in the order in which they are input to the model. The reader should refer to the figures and table described below to obtain a better picture and understanding of the KEMOD input structure. Figure 5-1 illustrates the listing of the sample problem input deck produced by subroutine PREDAT. Table 5-1 lists the input variables by code variable name, and describes their functions and input formats. Figure 5-2 contains a flow chart of the input variable sequence and options. The user will find these aids quite helpful in developing input decks for the program.

TABLE 5-1

CHEMICAL-DIFFUSION MODULE INPUT VARIABLE DESCRIPTION

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
1	TITLE	Label for each page of output.	(20A4)	May contain any user comments in columns 1-80.
2	GOAHED	Chemistry and Diffusion are to be computed after copying the input stream (YES) or (NO).	(40X,A4)	
3	FLXUNT	Units of emission data equal to MOLE or MASS fraction.	(40X,A4)	When FLXUNT ≠ MOLE, the molecular weights in WTMOLE are used in the conversion of fluxes to a molal basis.
4	NUMFLX	Number of species with area source emissions.	(40X,I10)	Maximum is seven species.
5	FLXTIM, FLXIN(1,I) ...,FLXIN (7,I)	Area source emission times and rates in mole or mass fraction-meters/minute.	(20X,F10.2, 4E12.4,/, 30X,3E12.4)	The first card has a time and emissions data for four species. The second card has emission data for three species. The pair of inputs may be repeated up to 200 times.
6	NPSFLX	Number of species with point source emissions.	(40X,I10)	Maximum of seven species.
7	TPASS(I), PTSR(1,I), ..., PTSR (7,I),VFR (1,I)..., VFR(5,I)	Point source emission times and mole fraction or mass fraction-meters entrained into the air parcel. Vertical distribution (0. to 1.) of the emissions with respect to the vertical mesh points.	(20X,F10.2, 4E12.4,/, 30X,3E12.4,/, 18X,5E12.4)	The first card has the time and emissions data for four species. The second card has emission data for three species. The third card has the vertical distribution. This trio of cards is then repeated up to 200 times.

TABLE 5-1 (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
8	DELT	Initial value of integration time step size in minutes.	(40X,F10.0)	1.E-3>DELT>1.E-5
9	BIGSTP	Maximum integration time step size in minutes.	(40X,F10.0)	8.>BIGSTP>4.
10	EPS	Integration error control criteria.	(40X,F10.0)	1.E-3>EPS>1.E-4
11	UPDINT	The time interval in minutes at which the program checks the various schedules of time-varying parameters for new data.	(40X,F10.0)	20.>UPDINT>10.
12	PRNTIN	Time interval at which printed output is desired in minutes.	(40X,A4)	Must be equal or multiple of UPDINT.
13	TIMOUT	The integration progress variables are to be printed (YES) or (NO).	(40X,A4)	The integration index, last time step size, last order, number of steps, number of functions, and number of Jacobians are printed.
14	SUNGEN	The photodissociation rates are to be generated internally (YES) or (NO).	(40X,A4)	A schedule of NO_2 and HCHO photolysis rates must be read in if SUNGEN ≠ YES.
15	HIRATE	The photodissociation rates vary with elevation (YES or NO).	(40X,A4)	
16	QCLOUD	A schedule of sky clearness ratios are input (YES) or (NO).	(40X,A4)	Clear sky photodissociation rates are used when QCLOUD = NO.

TABLE 5-1 (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
17	VTEMP	A schedule of atmospheric temperatures are input (YES) or (NO).	(40X,A4)	T = 25^{o}C is used when VTEMP = NO.
18	SPUNCH	The computed ground concentration histories are to be punched (YES) or (NO).	(40X,A4)	When SPUNCH = YES, the time concentration history is punched at each print interval in the format E12.4,/, 7(6E12.4,/).
19	NUMSTP	Maximum number of integration time steps.	(40X,I10)	NUMSTP should be equal to at least 75 steps per hour of real time simulation.
20	NOREAC	Number of chemical reactions	(40X,I10)	Equals 51 for the ERT mechanism.
21	NOSPEC	Total number species	(40X,I10)	Equal to 30 for the ERT mechanism.
22	NSTDY	Number of species approximated by the psuedo steady-state assumption or held constant.	(40X,I10)	Equal to 1 for the ERT mechanism.
23	NOSTAT	Number of vertical mesh points or cells.	(40X,I10)	$4 \leq NOSTAT \leq 5$. NOSTAT = 5 is recommended.
24	ZEE	Height of vertical mesh points in meters above the ground.	(40X,F10.0)	NOSTAT cards are read.
25	SPEC, WTMOLE, BCFLAG	Vectors of species names, molecular weights, and surface boundary condition codes.	(40X,A4,6X, F10.2,I10)	NOSPEC cards are read. See Section 5.1.3.
26	LOCFLX	A vector of species' indices corresponding to the area source emissions species.	(40X,I10)	NUMFLX cards read. See Section 5.1.4.

TABLE 5-1. (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
27	LOCPSF	A vector of species' indices corresponding to the point source emissions species.	(40X,I10)	NPSFLX cards used. See Section 5.1.4.
28	LOCDPF, DPRATE, DEPOWR	The index of a species with surface deposition, their deposition velocities in meters/min., and the exponent of the concentrations.	(40X,I10, 2F10.2)	The number cards must equal the number of species with BCFLAG = 1.
29	PPM	A two-dimensional array of initial concentrations of each species at each mesh point in ppm.	(20X,5E10.4)	NOSPEC cards read, with NOSTAT concentration for one species per card.
30	RATKON	A vector of rate constants in units consistent with the order of the reactions. (e.g., the units of a 2nd order reaction rate constant are ppm^{-1}-$minute^{-1}$).	(40X,F10.0)	NOREAC CARDS
31	REACT	An alphametric description of the chemical reaction mechanism.	(20A4)	NOREAC CARDS
32	RLAT	Latitude to be used in the zenith angle computation in degrees.	(40X,F10.0)	
33	RLONG	Longitude to be used in the zenith angle computation in degrees.	(40X,F10.0)	

TABLE 5-1 (CONTINUED)

Variable Number	Variable Name	Variable Function	FORTRAN Format	Remarks
34	TMZONE	Time zone of the region (e.g. 5. for New York and 8. for Los Angeles).	(40X,F10.0)	Also equal to the time difference between Greenwich, England, time and standard time in the modeling region.
35	JDATE	Date of day being simulated YRMODA (e.g., 760629 is June 29, 1976).	(40X,I10)	
36	NVRATE	Number of independent photodissociation rate arrays.	(40X,I10)	Maximum of two.
37	LOCVRT	The indices of the chemical reactions associated with the photodissociation rate arrays.	(40X,I10)	NVRATE cards
38	RATK1, RATK2	Arrays of the NO_2 and H_2CO photodissociation rates as a function of time and height.	(20X,5E12.4)	Required only if SUNGEN ≠ YES. RATK1 and RATK2 must be input on a time interval equal to UPDINT.
39	CLOUDT, CLOUDF	A table of sky clearness ratios as a function of time.	(40X,2F10.0)	Required only with QCLOUD = YES.
40	TMTEMP, TEMP	A table of temperatures (^{o}C) as a function of time.	(40X,2F10.0)	Required only with VTEMP = YES.
41	HTINV	A table of eddy diffusion coefficients (K_z) as a function of time and height.	(20X,F10.0, 10X,3F10.0,/, 20X,3F10.0)	

```
         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
   1     ERT CHEMICAL-DIFFUSION MODULE SIMULATION    --   ST. LOUIS -  6/29/76   (181)
   2     DO WE COMPUTE CHEMICAL DIFFUSION                 YES
   3     UNITS OF EMISSION FLUXES                         MOLE FRACTION
   4     NUMBER OF AREA SOURCE EMISSION SPECIE                     7
   5     AREA SOURCES    1          360.00  0.          0.          0.          0.
   6                                        0.          0.          0.
   7     AREA SOURCES    2          394.40  6.2948E-10  4.7461E-11  3.0947E-11  2.1613E-11
   8                                        1.5232E-11  3.9946E-09  3.4242E-11
   9     AREA SOURCES    3          420.00  6.3056E-10  4.7468E-11  3.0953E-11  2.1616E-11
  10                                        1.5238E-11  3.9951E-09  3.4242E-11

         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
  11     AREA SOURCES    4          421.46  1.0695E-09  3.1599E-10  2.1080E-10  9.0539E-11
  12                                        5.7753E-11  1.6214E-08  4.7693E-11
  13     AREA SOURCES    5          426.39  1.3987E-09  3.8370E-10  2.8401E-10  1.3581E-10
  14                                        8.4703E-11  2.3710E-08  6.0059E-11
  15     AREA SOURCES    6          480.00  1.2794E-09  3.3902E-10  2.4868E-10  1.2101E-10
  16                                        7.5869E-11  2.0861E-08  5.6217E-11
  17     AREA SOURCES    7          486.07  1.1568E-09  4.8803E-10  2.7318E-10  1.0276E-10
  18                                        6.2245E-11  1.6862E-08  5.0526E-11
  19     AREA SOURCES    8          496.41  7.0699E-10  2.2365E-10  9.5628E-11  3.4384E-11
  20                                        1.9921E-11  4.7868E-09  3.6179E-11

         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
  21     AREA SOURCES    9          511.58  5.2604E-10  6.1504E-11  3.4266E-11  2.8534E-11
  22                                        1.6236E-11  4.1231E-09  2.7948E-11
  23     AREA SOURCES   10          521.06  5.3719E-10  1.2717E-10  5.9833E-11  2.6795E-11
  24                                        1.6348E-11  4.1926E-09  2.8485E-11
  25     AREA SOURCES   11          537.08  3.8360E-10  1.2554E-10  5.1388E-11  2.6212E-11
  26                                        1.2376E-11  2.9728E-09  1.9811E-11
  27     AREA SOURCES   12          540.00  3.8517E-10  1.1836E-10  4.8767E-11  2.6215E-11
  28                                        1.2384E-11  2.9734E-09  1.9811E-11
  29     AREA SOURCES   13          551.01  5.4923E-10  9.9462E-10  4.6101E-10  1.4888E-10
  30                                        9.6169E-11  9.8887E-09  3.1663E-11

         1         11        21        31        41        51        61        71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
  31     AREA SOURCES   14          556.62  1.1345E-09  6.7646E-11  3.6272E-11  3.0139E-11
  32                                        2.0793E-11  4.2536E-09  6.2029E-11
  33     AREA SOURCES   15          564.13  2.4895E-10  4.7590E-11  1.8350E-11  2.2275E-11
  34                                        8.6010E-12  2.0085E-09  1.2472E-11
  35     AREA SOURCES   16          570.04  2.4895E-10  4.7590E-11  1.8350E-11  2.2275E-11
  36                                        8.6010E-12  2.0085E-09  1.2472E-11
  37     AREA SOURCES   17          571.63  4.7047E-10  2.2978E-10  6.0799E-11  1.0862E-10
  38                                        2.7733E-11  4.9042E-09  1.3762E-11
  39     AREA SOURCES   18          579.14  9.8934E-10  1.1668E-09  3.5381E-10  2.8772E-10
  40                                        6.8099E-11  1.0966E-08  1.9000E-11
```

Figure 5-1   Chemical-Diffusion Module Sample Problem Input Listing

```
      1         11        21        31        41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 41   AREA SOURCES  19            586.65  6.8719E-09 5.9080E-09 3.2800E-09 1.0595E-09
 42                                       6.5059E-10 1.9173E-07 1.8814E-10
 43   AREA SOURCES  20            589.06  1.4868E-09 3.3767E-09 1.2984E-09 2.4808E-10
 44                                       9.4978E-11 2.1869E-08 3.8482E-11
 45   AREA SOURCES  21            594.15  3.3096E-09 3.5885E-09 2.6686E-09 1.5946E-09
 46                                       1.0644E-09 1.3795E-07 1.3013E-10
 47   AREA SOURCES  22            598.58  6.3348E-09 1.2495E-08 7.3298E-09 3.0559E-09
 48                                       2.1032E-09 2.7206E-07 2.6124E-10
 49   AREA SOURCES  23            600.00  6.8837E-09 1.2722E-08 7.5204E-09 3.1374E-09
 50                                       2.1592E-09 2.8914E-07 2.7658E-10

      1         11        21        31        41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 51   AREA SOURCES  24            601.97  6.9314E-09 4.3280E-09 3.0233E-09 1.4255E-09
 52                                       8.6568E-10 2.4549E-07 2.2275E-10
 53   AREA SOURCES  25            610.88  6.9818E-10 6.5939E-10 2.0154E-10 1.8025E-10
 54                                       4.5322E-11 7.5543E-09 1.5773E-11
 55   AREA SOURCES  26            614.85  1.9784E-08 1.1513E-08 7.9774E-09 3.1679E-09
 56                                       2.0370E-09 5.8296E-07 6.4022E-10
 57   AREA SOURCES  27            619.79  6.9015E-10 3.8022E-10 9.8824E-11 1.7992E-10
 58                                       4.4960E-11 7.5380E-09 1.5773E-11
 59   AREA SOURCES  28            628.70  1.4073E-08 7.3394E-09 6.3035E-09 2.5720E-09
 60                                       1.9403E-09 6.3982E-07 3.2850E-10

      1         11        21        31        41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 61   AREA SOURCES  29            630.90  3.5109E-08 1.9592E-08 1.5709E-08 6.1848E-09
 62                                       4.5420E-09 1.4549E-06 9.1715E-10
 63   AREA SOURCES  30            637.61  6.9160E-09 1.1519E-08 7.4915E-09 3.8365E-09
 64                                       2.5417E-09 4.0971E-07 2.5992E-10
 65   AREA SOURCES  31            646.52  7.0907E-08 1.6750E-08 1.2083E-08 5.0369E-09
 66                                       3.6717E-09 1.0389E-06 2.9897E-09
 67   AREA SOURCES  32            646.94  8.5201E-08 3.5487E-08 2.6511E-08 1.1264E-08
 68                                       7.4444E-09 2.0575E-06 2.3277E-09
 69   AREA SOURCES  33            655.43  5.0874E-08 3.6667E-08 2.4756E-08 9.8677E-09
 70                                       6.4254E-09 1.9298E-06 1.5394E-09

      1         11        21        31        41         51         61         71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 71   AREA SOURCES  34            660.00  5.5525E-08 3.9251E-08 2.6834E-08 1.0684E-08
 72                                       7.0150E-09 2.1113E-06 1.6813E-09
 73   AREA SOURCES  35            663.44  2.1423E-08 1.5587E-08 1.0238E-08 5.0057E-09
 74                                       2.8840E-09 8.2212E-07 6.8713E-10
 75   AREA SOURCES  36            664.20  9.7319E-08 2.7934E-08 2.1000E-08 1.1056E-08
 76                                       6.3545E-09 2.4187E-06 8.6423E-09
 77   AREA SOURCES  37            672.81  8.9402E-08 3.9616E-08 2.6604E-08 1.2319E-08
 78                                       7.4717E-09 2.0123E-06 2.3221E-09
 79   AREA SOURCES  38            681.42  2.2408E-07 6.7122E-08 4.9568E-08 2.1947E-08
 80                                       1.4158E-08 3.5628E-06 5.0201E-09
```

Figure 5-1 (Continued)

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  81 AREA SOURCES  39                  681.90 1.2276E-08 9.4233E-09 5.0348E-09 3.9136E-09
  82                                          1.6197E-09 3.9667E-07 4.3259E-10
  83 AREA SOURCES  40                  685.72 1.4732E-07 4.4158E-08 3.4444E-08 1.4622E-08
  84                                          9.7926E-09 2.5547E-06 4.0982E-09
  85 AREA SOURCES  41                  690.03 2.4403E-07 7.2917E-08 5.6342E-08 2.4167E-08
  86                                          1.6213E-08 4.3681E-06 5.9780E-09
  87 AREA SOURCES  42                  691.13 3.0036E-08 1.6861E-08 7.5976E-09 7.1257E-09
  88                                          2.4762E-09 6.0171E-07 1.1920E-09
  89 AREA SOURCES  43                  694.33 7.7762E-08 4.1972E-08 2.4156E-08 9.7636E-09
  90                                          5.5584E-09 1.4500E-06 2.3799E-09

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
  91 AREA SOURCES  44                  698.64 6.6096E-07 1.0102E-07 6.9284E-08 3.1858E-08
  92                                          1.9275E-08 7.1032E-06 2.6708E-08
  93 AREA SOURCES  45                  700.36 4.7469E-08 3.9551E-08 1.9406E-08 9.9850E-09
  94                                          4.4210E-09 1.1573E-06 1.7010E-09
  95 AREA SOURCES  46                  702.94 2.6406E-07 1.0139E-07 7.0569E-08 3.0225E-08
  96                                          2.0506E-08 4.4130E-06 7.3346E-09
  97 AREA SOURCES  47                  707.25 6.0129E-08 2.0018E-08 1.5328E-08 9.4348E-09
  98                                          7.6152E-09 5.1955E-07 4.6735E-09
  99 AREA SOURCES  48                  709.59 1.0194E-06 1.6117E-07 1.2060E-07 5.5458E-08
 100                                          3.4766E-08 1.3522E-05 4.1097E-08

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 101 AREA SOURCES  49                  711.55 8.3940E-08 4.8356E-08 3.1299E-08 1.4111E-08
 102                                          1.1100E-08 1.0924E-06 4.6711E-09
 103 AREA SOURCES  50                  715.85 3.6132E-08 4.4707E-09 3.2550E-09 1.7789E-09
 104                                          1.7254E-09 3.7041E-07 2.5173E-09
 105 AREA SOURCES  51                  720.00 3.6127E-08 4.4706E-09 3.2550E-09 1.7789E-09
 106                                          1.7254E-09 3.7041E-07 2.5172E-09
 107 AREA SOURCES  52                  723.60 6.4366E-08 1.6453E-08 1.2999E-08 7.6082E-09
 108                                          6.9096E-09 4.3301E-07 5.3222E-09
 109 AREA SOURCES  53                  723.89 8.2130E-08 4.4793E-09 3.8673E-09 1.9805E-09
 110                                          1.5649E-09 7.6500E-07 4.5628E-09

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 111 AREA SOURCES  54                  727.64 1.6197E-07 3.0791E-08 2.6676E-08 1.1577E-08
 112                                          8.3109E-09 3.3317E-06 6.8794E-09
 113 AREA SOURCES  55                  727.72 1.2944E-07 3.5657E-08 2.8664E-08 1.1405E-08
 114                                          8.6244E-09 3.1642E-06 5.2915E-09
 115 AREA SOURCES  56                  731.38 2.9946E-08 1.2529E-08 5.6089E-09 2.4802E-09
 116                                          1.9029E-09 2.5645E-07 2.1766E-09
 117 AREA SOURCES  57                  731.85 2.0728E-08 1.5666E-08 7.7027E-09 3.6459E-09
 118                                          1.9100E-09 4.6731E-07 8.0803E-10
 119 AREA SOURCES  58                  735.13 1.9631E-07 4.7822E-08 3.3037E-08 1.5582E-08
 120                                          9.9254E-09 3.9886E-06 8.0359E-09
```

Figure 5-1 (Continued)

```
      1         11        21        31         41          51          61          71
CARD  +.........+.........+.........+..........+...........+...........+...........+..........
 121  AREA SOURCES  59                  735.97 4.6300E-08  1.0985E-07  4.5899E-08  8.7987E-09
 122                                           4.8330E-09  1.2811E-06  1.6610E-09
 123  AREA SOURCES  60                  738.88 8.1440E-08  4.2957E-08  2.8854E-08  1.3311E-08
 124                                           8.1882E-09  2.1326E-06  3.2988E-09
 125  AREA SOURCES  61                  740.10 4.7636E-08  2.0120E-08  1.2847E-08  4.9779E-09
 126                                           3.5494E-09  9.2125E-07  1.5318E-09
 127  AREA SOURCES  62                  742.63 2.2528E-08  1.2990E-08  6.5250E-09  2.7563E-09
 128                                           1.7477E-09  3.7916E-07  1.1440E-09
 129  AREA SOURCES  63                  744.22 6.7886E-08  2.0953E-08  1.5628E-08  5.6098E-09
 130                                           5.9397E-09  1.0349E-06  6.3957E-09

      1         11        21        31         41          51          61          71
CARD  +.........+.........+.........+..........+...........+...........+...........+..........
 131  AREA SOURCES  64                  746.38 5.1974E-08  9.6121E-09  5.5713E-09  1.4503E-09
 132                                           3.3554E-09  1.2986E-07  6.6724E-09
 133  AREA SOURCES  65                  748.35 1.7501E-08  7.1070E-09  5.1910E-09  2.3606E-09
 134                                           1.4522E-09  4.0337E-07  6.4837E-10
 135  AREA SOURCES  66                  750.13 5.5456E-08  1.0458E-08  6.0765E-09  2.1443E-09
 136                                           2.1263E-09  5.2400E-07  3.8953E-09
 137  AREA SOURCES  67                  752.47 2.9751E-08  8.5048E-09  6.4970E-09  2.5843E-09
 138                                           1.8640E-09  4.8754E-07  8.0619E-10
 139  AREA SOURCES  68                  753.87 4.2545E-08  2.4172E-08  1.8510E-08  6.9910E-09
 140                                           5.0765E-09  1.5675E-06  1.2504E-09

      1         11        21        31         41          51          61          71
CARD  +.........+.........+.........+..........+...........+...........+...........+..........
 141  AREA SOURCES  69                  756.60 1.1158E-07  4.1985E-08  3.2555E-08  1.1986E-08
 142                                           9.0309E-09  2.6394E-06  2.6261E-09
 143  AREA SOURCES  70                  757.62 7.7618E-08  2.4618E-08  1.6697E-08  5.7647E-09
 144                                           4.1363E-09  9.9341E-07  1.9096E-09
 145  AREA SOURCES  71                  760.72 2.6556E-08  1.2410E-08  1.1216E-08  4.4364E-09
 146                                           3.5042E-09  1.1621E-06  6.8794E-10
 147  AREA SOURCES  72                  761.37 2.2253E-09  2.4085E-10  9.9396E-11  7.9723E-11
 148                                           5.2463E-11  1.0590E-08  1.2042E-10
 149  AREA SOURCES  73                  764.85 9.7855E-09  3.8962E-09  3.0311E-09  1.1224E-09
 150                                           9.1493E-10  2.7167E-07  3.8702E-10

      1         11        21        31         41          51          61          71
CARD  +.........+.........+.........+..........+...........+...........+...........+..........
 151  AREA SOURCES  74                  780.00 1.0082E-08  4.0294E-09  3.1500E-09  1.1693E-09
 152                                           9.5131E-10  2.8343E-07  3.9404E-10
 153  AREA SOURCES  75                  780.14 1.6879E-08  4.7010E-09  3.7089E-09  1.4773E-09
 154                                           1.1489E-09  2.9758E-07  5.5979E-10
 155  AREA SOURCES  76                  786.01 1.2894E-08  5.8629E-09  4.2625E-09  1.5699E-09
 156                                           1.1814E-09  3.5647E-07  4.7925E-10
 157  AREA SOURCES  77                  804.21 1.8912E-08  5.0814E-09  3.6166E-09  1.3528E-09
 158                                           1.0023E-09  2.3522E-07  5.8661E-10
 159  AREA SOURCES  78                  808.68 4.2857E-09  6.7493E-10  3.5794E-10  1.1574E-10
 160                                           2.1933E-10  1.1440E-08  4.0399E-10
```

Figure 5-1 (Continued)

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 161 AREA SOURCES  79        809.03 2.0977E-09 2.0957E-10 9.1448E-11 7.0555E-11
 162                                4.8433E-11 1.0015E-08 1.1491E-10
 163 AREA SOURCES  80        817.74 2.1125E-09 5.0296E-10 1.9941E-10 7.0904E-11
 164                                4.8863E-11 1.0034E-08 1.1491E-10
 165 AREA SOURCES  81        818.65 1.5446E-08 6.6220E-09 3.9139E-09 1.1770E-09
 166                                8.1311E-10 2.0427E-07 4.1824E-10
 167 AREA SOURCES  82        826.81 1.0788E-08 2.2318E-09 1.6973E-09 6.9017E-10
 168                                5.0024E-10 1.1919E-07 3.2680E-10
 169 AREA SOURCES  83        828.28 3.0009E-09 1.6689E-09 7.6653E-10 1.4068E-10
 170                                1.0657E-10 2.7944E-08 1.5557E-10

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 171 AREA SOURCES  84        831.34 9.7101E-09 2.0080E-09 1.5118E-09 6.0814E-10
 172                                4.4745E-10 1.0672E-07 3.1587E-10
 173 AREA SOURCES  85        840.00 1.1121E-08 2.3653E-09 1.7899E-09 7.1755E-10
 174                                5.2452E-10 1.2570E-07 3.4617E-10
 175 AREA SOURCES  86        849.08 5.9993E-09 1.4448E-09 9.9115E-10 3.7468E-10
 176                                2.7190E-10 6.9699E-08 2.3467E-10
 177 AREA SOURCES  87        858.58 6.4817E-09 1.4557E-09 1.0764E-09 4.3601E-10
 178                                3.2000E-10 8.3000E-08 2.6381E-10
 179 AREA SOURCES  88        866.76 0.         0.         0.         0.
 180                                0.         0.         0.

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 181 AREA SOURCES  89        900.00 0.         0.         0.         0.
 182                                0.         0.         0.
 183 AREA SOURCES  90        960.00 0.         0.         0.         0.
 184                                0.         0.         0.
 185 AREA SOURCES  91       1020.00 0.         0.         0.         0.
 186                                0.         0.         0.
 187 LAST AREA SOURCES       -10.00
 188 LAST AREA SOURCES       -10.00
 189 NUMBER OF POINT SOURCE EMISSION SPECIE        7
 190 POINT SOURCES  1        709.80 0.         9.2240E-08 3.3810E-08 0.

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 191                                0.         0.         0.
 192                1    1.2828E-01 2.9127E-01 3.6558E-01 2.1488E-01 0.
 193 POINT SOURCES  2        711.40 7.6917E-07 0.         0.         0.
 194                                0.         0.         1.9874E-06
 195                2    1.7865E-01 2.0977E-01 4.5516E-01 1.5641E-01 0.
 196 POINT SOURCES  3        711.48 4.4139E-08 3.2283E-09 6.9287E-09 4.5684E-10
 197                                1.3857E-09 1.2690E-06 3.9656E-08
 198                3    1.3890E-01 2.2231E-01 4.2099E-01 2.1781E-01 0.
 199 POINT SOURCES  4        711.50 4.4175E-08 3.2310E-09 6.9344E-09 4.5722E-10
 200                                1.3869E-09 1.2700E-06 3.9689E-08
```

Figure 5-1 (Continued)

```
       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 201                   4  1.3889E-01  2.2231E-01  4.2097E-01  2.1783E-01  0.
 202   POINT SOURCES   5      719.64  0.          5.3696E-08  1.9682E-08  0.
 203                                  0.          0.          0.
 204                   5  1.3251E-01  3.0197E-01  3.8773E-01  1.7778E-01  0.
 205   POINT SOURCES   6      721.87  0.          1.4858E-07  0.          2.0688E-08
 206                                  0.          0.          0.
 207                   6  1.4159E-01  3.2003E-01  4.0849E-01  1.2988E-01  0.
 208   POINT SOURCES   7      721.87  0.          2.2484E-11  0.          5.2169E-12
 209                                  0.          0.          0.
 210                   7  1.4159E-01  3.2003E-01  4.0849E-01  1.2988E-01  0.

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 211   POINT SOURCES   8      722.70  0.          1.5356E-07  5.6287E-08  0.
 212                                  0.          0.          0.
 213                   8  1.5620E-01  3.0831E-01  3.4087E-01  1.9462E-01  0.
 214   POINT SOURCES   9      722.70  0.          2.8933E-08  1.0605E-08  0.
 215                                  0.          0.          0.
 216                   9  1.5333E-01  3.0859E-01  3.4212E-01  1.9596E-01  0.
 217   POINT SOURCES  10      725.51  0.          8.1482E-08  2.9866E-08  0.
 218                                  0.          0.          0.
 219                  10  1.3294E-01  3.0126E-01  3.8733E-01  1.7847E-01  0.
 220   POINT SOURCES  11      728.18  0.          0.          0.          0.

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 221                                  0.          0.          6.4126E-07
 222                  11  1.5995E-01  2.9466E-01  3.2098E-01  2.2442E-01  0.
 223   POINT SOURCES  12      728.42  6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13
 224                                  9.4022E-13  1.5857E-10  2.4485E-12
 225                  12  1.6146E-01  3.1331E-01  3.4446E-01  1.8077E-01  0.
 226   POINT SOURCES  13      728.42  6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13
 227                                  9.4022E-13  1.5857E-10  2.4485E-12
 228                  13  1.6146E-01  3.1331E-01  3.4446E-01  1.8077E-01  0.
 229   POINT SOURCES  14      728.42  6.8132E-10  9.4022E-13  9.4022E-13  3.1341E-13
 230                                  9.4022E-13  1.5857E-10  2.4485E-12

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 231                  14  1.6146E-01  3.1331E-01  3.4446E-01  1.8077E-01  0.
 232   POINT SOURCES  15      728.69  0.          1.1246E-08  4.1222E-09  0.
 233                                  0.          0.          0.
 234                  15  1.3385E-01  3.0337E-01  3.8761E-01  1.7518E-01  0.
 235   POINT SOURCES  16      728.95  0.          0.          0.          0.
 236                                  0.          0.          3.6272E-08
 237                  16  1.5899E-01  3.0905E-01  3.3927E-01  1.9269E-01  0.
 238   POINT SOURCES  17      728.97  0.          0.          0.          0.
 239                                  0.          0.          2.4949E-08
 240                  17  1.6733E-01  3.0824E-01  3.3558E-01  1.8885E-01  0.
```

Figure 5-1 (Continued)

```
       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 241   POINT SOURCES 18        729.04 0.          0.          0.          0.
 242                                  0.          0.          1.0422E-06
 243                 18    1.6734E-01 3.0825E-01  3.3558E-01  1.8882E-01  0.
 244   POINT SOURCES 19        729.29 0.          0.          0.          0.
 245                                  0.          0.          2.4811E-07
 246                 19    1.6385E-01 3.0867E-01  3.3716E-01  1.9031E-01  0.
 247   POINT SOURCES 20        729.30 0.          0.          0.          3.6492E-09
 248                                  0.          0.          0.
 249                 20    2.6556E-01 2.1556E-01  3.3444E-01  1.8443E-01  0.
 250   POINT SOURCES 21        729.30 0.          0.          0.          1.8246E-08

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 251                                  0.          0.          0.
 252                 21    2.6556E-01 2.1556E-01  3.3444E-01  1.8443E-01  0.
 253   POINT SOURCES 22        729.30 0.          0.          0.          9.2889E-09
 254                                  0.          0.          0.
 255                 22    1.5009E-01 2.9552E-01  3.2529E-01  2.2910E-01  0.
 256   POINT SOURCES 23        729.30 0.          0.          0.          3.9312E-08
 257                                  0.          0.          0.
 258                 23    2.5326E-01 2.0558E-01  3.1895E-01  2.2221E-01  0.
 259   POINT SOURCES 24        729.30 0.          0.          0.          3.5663E-08
 260                                  0.          0.          0.

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 261                 24    1.5162E-01 2.9538E-01  3.2462E-01  2.2838E-01  0.
 262   POINT SOURCES 25        729.30 0.          0.          0.          3.6492E-09
 263                                  0.          0.          0.
 264                 25    2.5326E-01 2.0558E-01  3.1895E-01  2.2221E-01  0.
 265   POINT SOURCES 26        729.30 0.          0.          0.          2.1564E-09
 266                                  0.          0.          0.
 267                 26    1.4824E-01 2.9569E-01  3.2610E-01  2.2997E-01  0.
 268   POINT SOURCES 27        730.04 2.4348E-07  8.9601E-10  8.9601E-10  2.9867E-10
 269                                  8.9601E-10  5.3334E-08  3.6797E-07
 270                 27    1.6425E-01 1.7565E-01  4.2315E-01  2.3695E-01  0.

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 271   POINT SOURCES 28        730.11 8.3123E-08  2.2942E-11  2.2942E-11  7.6474E-12
 272                                  2.2942E-11  3.8692E-09  5.9744E-11
 273                 28    1.7395E-01 1.6796E-01  4.4093E-01  2.1716E-01  0.
 274   POINT SOURCES 29        730.43 5.7404E-08  2.1125E-10  2.1125E-10  7.0417E-11
 275                                  2.1125E-10  1.2574E-08  8.6755E-08
 276                 29    1.7551E-01 1.6628E-01  4.4360E-01  2.1461E-01  0.
 277   POINT SOURCES 30        730.59 5.2771E-08  1.9420E-10  1.9420E-10  6.4732E-11
 278                                  1.9420E-10  1.1559E-08  7.9752E-08
 279                 30    1.7647E-01 1.6521E-01  4.4524E-01  2.1308E-01  0.
 280   POINT SOURCES 31        731.29 4.7517E-09  4.3715E-12  4.3715E-12  1.4572E-12
```

Figure 5-1 (Continued)

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+........
 281                                4.3715E-12  4.3368E-10  2.0404E-08
 282               31    1.5967E-01 2.0883E-01  4.7483E-01  1.5667E-01  0.
 283 POINT SOURCES 32        731.32 2.8522E-07  7.8721E-11  7.8721E-11  2.6240E-11
 284                                7.8721E-11  1.3276E-08  2.0500E-10
 285               32    1.4429E-01 2.0637E-01  4.3343E-01  2.1591E-01  0.
 286 POINT SOURCES 33        731.63 4.7094E-09  4.3327E-12  4.3327E-12  1.4442E-12
 287                                4.3327E-12  4.2982E-10  2.0223E-08
 288               33    1.5953E-01 2.0890E-01  4.7459E-01  1.5698E-01  0.
 289 POINT SOURCES 34        732.00 2.8082E-07  7.7508E-11  7.7508E-11  2.5836E-11
 290                                7.7508E-11  1.3072E-08  2.0184E-10

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+........
 291               34    1.4113E-01 2.1599E-01  4.2728E-01  2.1560E-01  0.
 292 POINT SOURCES 35        733.89 0.          0.          0.          8.0997E-07
 293                                0.          1.8246E-05  0.
 294               35    1.6241E-01 3.0735E-01  3.3717E-01  1.9307E-01  0.
 295 POINT SOURCES 36        734.71 8.9666E-08  3.2997E-10  3.2997E-10  1.0999E-10
 296                                3.2997E-10  1.9641E-08  5.0286E-07
 297               36    1.7464E-01 1.6727E-01  4.4244E-01  2.1565E-01  0.
 298 POINT SOURCES 37        735.11 0.          0.          0.          0.
 299                                0.          0.          1.7949E-08
 300               37    1.6668E-01 2.1072E-01  4.8124E-01  1.4136E-01  0.

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+........
 301 POINT SOURCES 38        735.15 6.9893E-10  1.0128E-10  8.3046E-09  4.7262E-10
 302                                0.          3.6456E-10  5.0236E-08
 303               38    1.5875E-01 3.1832E-01  3.6481E-01  1.5813E-01  0.
 304 POINT SOURCES 39        735.39 0.          3.6550E-08  0.          5.0892E-09
 305                                0.          0.          0.
 306               39    1.5077E-01 2.9744E-01  3.2587E-01  2.2591E-01  0.
 307 POINT SOURCES 40        735.58 0.          1.7957E-09  6.5819E-10  0.
 308                                0.          0.          0.
 309               40    1.5427E-01 3.1039E-01  3.4256E-01  1.9278E-01  0.
 310 POINT SOURCES 41        735.58 0.          2.6442E-08  9.6920E-09  0.

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+........
 311                                0.          0.          0.
 312               41    1.5427E-01 3.1039E-01  3.4256E-01  1.9278E-01  0.
 313 POINT SOURCES 42        735.58 0.          1.2432E-09  4.5569E-10  0.
 314                                0.          0.          0.
 315               42    1.5427E-01 3.1039E-01  3.4256E-01  1.9278E-01  0.
 316 POINT SOURCES 43        735.58 0.          5.0280E-08  1.8429E-08  0.
 317                                0.          0.          0.
 318               43    1.5427E-01 3.1039E-01  3.4256E-01  1.9278E-01  0.
 319 POINT SOURCES 44        735.58 0.          4.4746E-08  1.6401E-08  0.
 320                                0.          0.          0.
```

Figure 5-1 (Continued)

```
     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 321               44    1.5814E-01  3.1004E-01  3.4086E-01  1.9096E-01  0.
 322 POINT SOURCES 45        735.97  6.3644E-10  9.2221E-13  7.5621E-11  4.3036E-12
 323                                 0.          3.3197E-10  4.5745E-08
 324               45    1.5917E-01  3.1953E-01  3.6474E-01  1.5656E-01  0.
 325 POINT SOURCES 46        736.41  1.2917E-08  9.3005E-12  9.3005E-12  3.1002E-12
 326                                 9.3005E-12  1.5685E-09  2.4220E-11
 327               46    1.6205E-01  3.1071E-01  3.3899E-01  1.8825E-01  0.
 328 POINT SOURCES 47        737.04  1.6074E-08  1.1573E-11  1.1573E-11  3.8576E-12
 329                                 1.1573E-11  1.9518E-09  3.0138E-11
 330               47    1.7613E-01  1.6564E-01  4.4498E-01  2.1325E-01  0.

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 331 POINT SOURCES 48        737.35  1.7640E-08  1.2701E-11  1.2701E-11  4.2336E-12
 332                                 1.2701E-11  2.1420E-09  3.3074E-11
 333               48    1.7585E-01  1.6596E-01  4.4451E-01  2.1369E-01  0.
 334 POINT SOURCES 49        737.51  1.3700E-08  9.8642E-12  9.8642E-12  3.2881E-12
 335                                 9.8642E-12  1.6636E-09  2.5688E-11
 336               49    1.7077E-01  3.0615E-01  3.3291E-01  1.9017E-01  0.
 337 POINT SOURCES 50        739.81  4.7400E-09  3.4128E-12  3.4128E-12  1.1376E-12
 338                                 3.4128E-12  5.7556E-10  4.4437E-13
 339               50    1.5421E-01  3.3138E-01  3.7875E-01  1.3567E-01  0.
 340 POINT SOURCES 51        740.23  8.3047E-08  5.9794E-11  5.9794E-11  1.9931E-11

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 341                                 5.9794E-11  1.0084E-08  1.5571E-10
 342               51    1.5144E-01  1.9266E-01  4.0274E-01  2.5316E-01  0.
 343 POINT SOURCES 52        740.23  6.3142E-08  0.          0.          0.
 344                                 0.          0.          0.
 345               52    1.6552E-01  3.0916E-01  3.3587E-01  1.8945E-01  0.
 346 POINT SOURCES 53        740.32  4.4604E-09  3.2115E-12  3.2115E-12  1.0705E-12
 347                                 3.2115E-12  5.4162E-10  8.3633E-12
 348               53    1.5286E-01  3.3208E-01  3.7954E-01  1.3552E-01  0.
 349 POINT SOURCES 54        740.60  4.3951E-09  3.1645E-12  3.1645E-12  1.0548E-12
 350                                 3.1645E-12  5.3369E-10  4.1204E-13

     1         11        21        31        41        51        61        71
CARD +.........+.........+.........+.........+.........+.........+.........+.........
 351               54    1.4938E-01  3.3226E-01  3.8140E-01  1.3696E-01  0.
 352 POINT SOURCES 55        741.83  6.9778E-08  0.          0.          0.
 353                                 0.          0.          0.
 354               55    1.6375E-01  3.0930E-01  3.3662E-01  1.9033E-01  0.
 355 POINT SOURCES 56        741.88  0.          0.          0.          0.
 356                                 0.          0.          1.1199E-07
 357               56    1.5627E-01  3.3154E-01  3.7767E-01  1.3452E-01  0.
 358 POINT SOURCES 57        742.24  1.9817E-08  5.4696E-11  5.4696E-11  1.8232E-11
 359                                 5.4696E-11  2.1705E-09  7.4542E-08
 360               57    1.7204E-01  1.8905E-01  4.8182E-01  1.5709E-01  0.
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 361  POINT SOURCES 58        742.30  5.7709E-08  0.          0.          0.
 362                                  0.          0.          2.6396E-11
 363                58    1.6635E-01  3.1006E-01  3.3670E-01  1.8690E-01  0.
 364  POINT SOURCES 59        742.31  1.0225E-08  3.7627E-11  3.7627E-11  1.2542E-11
 365                                  3.7627E-11  2.2397E-09  6.1438E-08
 366                59    1.4436E-01  3.3200E-01  3.8401E-01  1.3963E-01  0.
 367  POINT SOURCES 60        742.37  1.3872E-08  5.1049E-11  5.1049E-11  1.7016E-11
 368                                  5.1049E-11  3.0386E-09  8.3353E-08
 369                60    1.6622E-01  3.3448E-01  3.7935E-01  1.1995E-01  0.
 370  POINT SOURCES 61        742.38  2.3494E-08  6.4843E-12  6.4843E-12  2.1614E-12

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 371                                  6.4843E-12  1.0936E-09  1.6886E-11
 372                61    1.7287E-01  1.8789E-01  4.8306E-01  1.5618E-01  0.
 373  POINT SOURCES 62        742.51  1.5186E-08  5.5887E-11  5.5887E-11  1.8629E-11
 374                                  5.5887E-11  3.3266E-09  9.1252E-08
 375                62    1.7186E-01  1.8926E-01  4.8146E-01  1.5741E-01  0.
 376  POINT SOURCES 63        742.51  2.3189E-08  6.4001E-12  6.4001E-12  2.1334E-12
 377                                  6.4001E-12  1.0794E-09  1.6667E-11
 378                63    1.7289E-01  1.8789E-01  4.8308E-01  1.5615E-01  0.
 379  POINT SOURCES 64        742.70  0.          0.          0.          0.
 380                                  0.          0.          1.3493E-07

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 381                64    1.6395E-01  3.3004E-01  3.7344E-01  1.3257E-01  0.
 382  POINT SOURCES 65        751.10  1.2966E-08  1.1260E-11  1.1260E-11  3.7533E-12
 383                                  1.1260E-11  1.5632E-09  6.6436E-10
 384                65    1.7301E-01  1.7207E-01  4.3993E-01  2.1498E-01  0.
 385  POINT SOURCES 66        751.10  1.2966E-08  1.1260E-11  1.1260E-11  3.7533E-12
 386                                  1.1260E-11  1.5632E-09  6.6436E-10
 387                66    1.7301E-01  1.7207E-01  4.3993E-01  2.1498E-01  0.
 388  POINT SOURCES 67        754.31  5.4798E-09  3.9455E-12  3.9455E-12  1.3152E-12
 389                                  3.9455E-12  6.6540E-10  1.0274E-11
 390                67    1.6241E-01  3.1530E-01  3.4066E-01  1.8162E-01  0.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 391  POINT SOURCES 68        760.71  3.2740E-08  1.5862E-11  1.5862E-11  5.2875E-12
 392                                  1.5862E-11  1.6970E-09  4.8280E-09
 393                68    1.7791E-01  1.7212E-01  4.5038E-01  1.9959E-01  0.
 394  POINT SOURCES 69        766.27  0.          3.4072E-09  5.4929E-11  1.5794E-09
 395                                  0.          0.          0.
 396                69    1.4929E-01  2.9969E-01  3.2750E-01  2.2352E-01  0.
 397  POINT SOURCES 70        768.67  5.5572E-08  5.8430E-11  5.8430E-11  1.9477E-11
 398                                  5.8430E-11  2.6085E-09  2.0007E-07
 399                70    1.7277E-01  1.7219E-01  4.3982E-01  2.1522E-01  0.
 400  POINT SOURCES 71        774.38  0.          6.2820E-10  2.3026E-10  0.
```

Figure 5-1 (Continued)

```
         1         11         21         31         41         51         61         71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
 401                                     0.          0.          0.
 402                     71   1.5718E-01  3.0683E-01  3.3990E-01  1.9409E-01  0.
 403     POINT SOURCES 72         800.86  1.8367E-09  5.0694E-12  5.0694E-12  1.6898E-12
 404                                      5.0694E-12  2.0117E-10  1.2498E-09
 405                     72   1.6603E-01  3.1068E-01  3.3514E-01  1.8815E-01  0.
 406     POINT SOURCES 73         821.76  2.1268E-09  2.2362E-12  2.2362E-12  7.4539E-13
 407                                      2.2362E-12  9.9828E-11  2.0964E-10
 408                     73   1.6694E-01  3.2300E-01  3.4218E-01  1.6788E-01  0.
 409     POINT SOURCES 74         821.87  2.1023E-09  2.2104E-12  2.2104E-12  7.3680E-13
 410                                      2.2104E-12  9.8678E-11  2.0723E-10

         1         11         21         31         41         51         61         71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
 411                     74   1.6693E-01  3.2297E-01  3.4217E-01  1.6793E-01  0.
 412     POINT SOURCES 75         822.07  2.0650E-09  2.1712E-12  2.1712E-12  7.2375E-13
 413                                      2.1712E-12  9.6930E-11  2.0355E-10
 414                     75   1.6692E-01  3.2294E-01  3.4216E-01  1.6798E-01  0.
 415     POINT SOURCES 76         822.18  2.0411E-09  2.1461E-12  2.1461E-12  7.1536E-13
 416                                      2.1461E-12  9.5806E-11  2.0119E-10
 417                     76   1.6691E-01  3.2291E-01  3.4216E-01  1.6803E-01  0.
 418     LAST PTSR               -100.00
 419     LAST PTSR
 420     INITIAL INTEGRATION TIME STEP SIZE          1.0E-04     (MINUTES)

         1         11         21         31         41         51         61         71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
 421     MAXIMUM INTEGRATION TIME STEP SIZE      6.0                 (MINUTES)
 422     INTEGRATION ERROR CONTROL CRITERIA          1.0E-03
 423     UPDATE INTERVAL                         15.0                (MINUTES)
 424     PRINT INTERVAL                          30.0                (MINUTES)
 425     DO WE OUTPUT INTEGRATION PARAMETERS?    YES
 426     DO WE GENERATE PHOTODISSOCIATION RATES? YES
 427     SHOULD THEY VARY WITH ELEVATION?        YES
 428     DO WE HAVE SKY CLEARNESS RATIOS INPUT?  YES
 429     DO WE HAVE TEMPERATURES INPUT?          YES
 430     SHOULD WE PUNCH GROUND CONCENTRATIONS?  NO

         1         11         21         31         41         51         61         71
CARD     +.........+.........+.........+.........+.........+.........+.........+.........
 431     MAXIMUM NUMBER OF INTEGRATION STEPS               1000
 432     NUMBER OF CHEMICAL REACTIONS                        51
 433     NUMBER OF SPECIE                                    30
 434     NUMBER OF SPECIE TREATED AS CONSTANTS                1
 435     NUMBER OF VERTICAL MESH POINTS                       5
 436     ELEVATION OF FIRST  VERTICAL MESH POINT 0.0               (METERS ABOVE SURFACE)
 437     ELEVATION OF SECOND VERTICAL MESH POINT 120.0
 438     ELEVATION OF THIRD  VERTICAL MESH POINT 300.0
 439     ELEVATION OF FOURTH VERTICAL MESH POINT 600.0
 440     ELEVATION OF FIFTH  VERTICAL MESH POINT 1200.0
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 441  SPECIES NAME, MOLE WT, AND BOUND COND   NO        30.
 442                                 2        NO2       46.
 443                                 3        O3        46.                 1
 444                                 4        HONO      47.
 445                                 5        CO        28.
 446                                 6        HCHO      30.
 447                                 7        RCHO      58.
 448                                 8        OLEF      42.
 449                                 9        PA        58.
 450                                10        NO3       62.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 451                                11        HO2       33.
 452                                12        PAN       121.
 453                                13        AR        92.
 454                                14        ARCO      156.
 455                                15        RO2       89.
 456                                16        RCO3      103.
 457                                17        AO2       74.
 458                                18        PAO2      89.
 459                                19        D         74.
 460                                20        HNO4      79.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 461                                21        ARO       123.
 462                                22        AROH      107.
 463                                23        AO        58.
 464                                24        OH        17.
 465                                25        O         16.
 466                                26        PAO       73.
 467                                27        N2O5      108.
 468  SO2 WITH DEPOSITION           28        SO2       64.                 1
 469                                29        SO4       96.                 1
 470                                30        H2O       18.

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 471  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 1          1
 472  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 2          9
 473  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 3          8
 474  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 4         13
 475  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 5          6
 476  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 6          5
 477  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 7         28
 478  SPECIE INDEX FOR POINT SOURCE FLUX NO. 1          1
 479  SPECIE INDEX FOR POINT SOURCE FLUX NO. 2          9
 480  SPECIE INDEX FOR POINT SOURCE FLUX NO. 3          8
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
481   SPECIE INDEX FOR POINT SOURCE FLUX NO. 4          13
482   SPECIE INDEX FOR POINT SOURCE FLUX NO. 5           6
483   SPECIE INDEX FOR POINT SOURCE FLUX NO. 6           5
484   SPECIE INDEX FOR POINT SOURCE FLUX NO. 7          28
485   DEPOSITION SPECIE INDEX,VELOCITY,EXP.             3    .3      1.0
486   DEPOSITION SPECIE INDEX,VELOCITY,EXP.            28    .48     1.0
487   DEPOSITION SPECIE INDEX,VELOCITY,EXP.            29    .18     1.0
488   NO     IC              3.0E-03   2.5E-03   1.5E-03   1.5E-03   1.5E-03
489   NO2       COMPUTED
490   O3     IC          .029      .043      .058      .058      .058

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
491   HONO      COMPUTED
492   CO     IC          .133      .110      .066      .066      .066
493   HCHO   IC              4.0E-04   3.4E-04   2.0E-04   2.0E-04   2.0F-04
494   RCHO   IC              2.0E-04   1.7E-04   1.0E-04   1.0E-04   1.0E-04
495   OLEF   IC              6.0E-04  5.10E-04   3.0E-04   3.0E-04   3.0E-04
496   PA     IC              3.0F-03  2.54E-03   1.5E-03   1.5E-03   1.5E-03
497   NO3       COMPUTED
498   HO2    IC              1.0E-06   1.0E-06   1.0E-06   1.0E-06   1.0E-06
499   PAN       COMPUTED
500   AR     IC              6.0E-04  5.10E-04   3.0E-04   3.0E-04   3.0E-04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
501   ARCO      COMPUTED
502   RO2       COMPUTED
503   RCO3      COMPUTED
504   AO2       COMPUTED
505   PAO2      COMPUTED
506   O         COMPUTED
507   HNO4      COMPUTED
508   ARO       COMPUTED
509   AROH      COMPUTED
510   AO        COMPUTED

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
511   OH     IC               1.E-08    1.E-08    1.E-08    1.E-08    1.E-08
512   O         COMPUTED
513   PAO       COMPUTED
514   N2O5      COMPUTED
515   SO2                    3.2E-03   .28E-03   1.6E-03   1.6E-03   1.6E-03
516   SO3                    3.2E-04   .28E-03   1.6E-03   1.6E-03   1.6E-03
517   H2O    IC              1.7E 04   1.7E 04   1.7E 04   1.7E 04   1.7E 04
518   RATE CONSTANT OF REACTION NO. 1                  COMPUTED FROM SOLAR
519             REACTION NO.        2          4.12E 06  1/MIN
520             REACTION NO.        3          2.50E 01
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  521           REACTION NO.        4         2.20E-09
  522           REACTION NO.        5         1.40E-03
  523           REACTION NO.        6                   COMPUTED FROM SOLAR
  524           REACTION NO.        7         1.50E 04
  525           REACTION NO.        8         1.50E 04
  526           REACTION NO.        9         4.40E 02
  527           REACTION NO.        10        1.20E 04
  528           REACTION NO.        11        1.70E 03
  529           REACTION NO.        12        8.40E 03
  530           REACTION NO.        13        5.00E-02

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  531           REACTION NO.        14        2.80E 04
  532           REACTION NO.        15        9.30E 02
  533           REACTION NO.        16        5.00E-05
  534           REACTION NO.        17        2.40E 01
  535           REACTION NO.        18        3.70E 04
  536           REACTION NO.        19        2.90E 04
  537           REACTION NO.        20        4.10E 05
  538           REACTION NO.        21                  COMPUTED FROM TEMPERATURE
  539           REACTION NO.        22        1.90E-02
  540           REACTION NO.        23        2.30E 04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  541           REACTION NO.        24        5.20E 03
  542           REACTION NO.        25        2.90E 04
  543           REACTION NO.        26        3.80E 03
  544           REACTION NO.        27        2.90E 04
  545           REACTION NO.        28        2.60E 03
  546           REACTION NO.        29        1.40E 05      1/MIN
  547           REACTION NO.        30        2.90E 04
  548           REACTION NO.        31        6.70E 04      1/MIN
  549           REACTION NO.        32        2.30E 04
  550           REACTION NO.        33        2.20E 04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  551           REACTION NO.        34                  COMPUTED FROM SOLAR
  552           REACTION NO.        35        2.90E 04
  553           REACTION NO.        36        1.70E 04
  554           REACTION NO.        37                  COMPUTED FROM TEMPERATURE
  555           REACTION NO.        38                  COMPUTED FROM SOLAR
  556           REACTION NO.        39        2.10E 04
  557           REACTION NO.        40        4.20E 03
  558           REACTION NO.        41        2.24E 04
  559           REACTION NO.        42        5.02E 04
  560           REACTION NO.        43        2.90E 04
```

Figure 5-1 (Continued)

```
        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 561              REACTION NO.        44          5.60E 03
 562              REACTION NO.        45          2.90E 04
 563                           46              8.4
 564                           47             1.76E 03
 565                           48              3.0E-02
 566                           49              2.9E 03
 567                           50              8.0
 568                           51              8.0
 569      1.NO2  +   HV               =   1.O    +  1.NO
 570      1.O    +    O2   +     M    =   1.O3   +    M

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 571      1.O3   +  1.NO              =   1.NO2  +    O2
 572      1.NO   +  1.NO2  +  1.H2O   =   2.HONO
 573      2.HONO                      =   1.NO   +  1.NO2  +  1.H2O
 574      1.HONO +   HV               =   1.OH   +  1.NO
 575      1.OH   +  1.NO   +     M    =   1.HONO +    M
 576      1.OH   +  1.NO2  +     M    =   1.HNO3 +    M
 577      1.OH   +  1.CO              =   1.HO2  +   CO2
 578      1.HO2  +  1.NO              =   1.NO2  +  1.OH
 579      1.HO2  +  1.NO2             =   1.HNO4
 580      2.HO2                       =     H2O2 +   O2

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 581      1.NO2  +  1.O3              =   1.NO3  +    O2
 582      1.NO3  +  1.NO              =   2.NO2
 583      1.NO3  +  1.NO2             =   1.N2O5
 584      1.N2O5 +  1.H2O             =   2.HNO3
 585      1.N2O5                      =   1.NO3  +  1.NO2
 586      1.OH   +  1.OLEF            =   1.AO2
 587      1.AO2  +  1.NO              =   1.NO2  +  1.AO
 588      1.AO   +    O2              =   1.RCHO +  1.HCHO +  1.HO2
 589      1.HNO4                      =   1.HO2  +  1.NO2
 590      1.O3   +  1.OLEF            =   .5HCHO +.5RCHO +.25HO2 +.25RCO3 +.5OH +.5O

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 591      1.O    +  1.NO2             =   1.NO3  +  .5HCHO +  .5RCHO
 592      1.O    +  1.OLEF            =   .3EPOX +  .3RCHO +  .4HO2  +  .4RO2
 593      1.O    +  1.NO              =   1.NO2  +  .5HCHO +  .5RCHO
 594      1.OH   +  1.PA              =   1.H2O  +  1.PAO2
 595      1.PAO2 +  1.NO              =   1.NO2  + .85PAO  + .15RO2
 596      1.PAO2 +  1.NO              =    NTRA
 597      1.PAO                       =   1.RO2  +  .5HCHO +  .5RCHO
 598      1.RO2  +  1.NO              =   1.NO2  +  1.PAO
 599      1.PAO  +    O2              =   .5KET  +  .5RCHO +  1.HO2
 600      1.PAO  +  1.NO2             =  .85NTRA +  .15RCHO +  .15HONO
```

Figure 5-1 (Continued)

```
      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 601    1.OH   +  1.RCHO            =   1.RCO3 +  1.H2O
 602    1.RCHO +    HV              =   1.RO2  +  1.HO2  +  1.CO
 603    1.RCO3 +  1.NO              =   1.RO2   +  1.NO2  +   CO2
 604    1.RCO3 +  1.NO2             =   1.PAN
 605    1.PAN                       =   1.RCO3 +  1.NO2
 606    1.HCHO +    HV              =  .67HO2  +  1.CO
 607    1.HCHO +  1.OH              =   1.H2O  +  1.HO2  +  1.CO
 608    1.RO2  +  1.HO2             =     RO2H +    O2
 609    1.AR   +  1.OH              =   1.AROH +  1.HO2
 610    1.AROH +  1.OH              =   1.ARCO

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 611    1.ARCO +  1.NO              =   1.NO2  +  1.RCHO
 612    1.AR   +  1.OH              =   1.H2O  +  1.ARO
 613    1.ARO  +  1.NO              =   1.NO2  +  1.HO2  +    HALO
 614    1.SO2  +  1.O               =   1.SO4
 615    1.SO2  +  1.OH              =   1.SO4
 616    1.SO2  +  1.HO2             =   1.SO4  +  1.OH
 617    1.SO2  +  1.O               =   1.SO4  +  1.RCHO
 618    1.SO2  +  1.RO2             =   1.SO4  +  1.PAO
 619    1.SO2  +  1.AO2             =   1.SO4  +  1.AO
 620  LATITUDE                                38.6

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 621  LONGITUDE                               90.2
 622  TIME ZONE                               6.0
 623  DATE   (YRMODA)                              760629
 624  NUMBER OF TIME-VARYING REACTION RATES             2
 625  REACTION NO. OF 1ST RATE CONSTANT ARRAY         1
 626  REACTION NO. OF 2ND RATE CONSTANT ARRAY         38
 627  SKY CLEARNESS RATIOS (TIME & CLEAR)     360.        .549
 628                                          420.        .703
 629                                          480.        .758
 630                                          540.        .725

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 631                                          600.        .695
 632                                          660.        .545
 633                                          720.        .568
 634                                          750.        .784
 635                                          765.        .890
 636                                          780.        1.00
 637                                          810.        .86
 638                                          840.        .727
 639                                          870.        .85
 640                                          900.        .975
```

Figure 5-1 (Continued)

```
       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+........
  641                                          960.       1.00
  642  LAST                                    -10.
  643  SURFACE TEMPERATURES                      360.     21.50
  644                                            420.     22.41
  645                                            480.     24.39
  646                                            540.     26.46
  647                                            600.     27.67
  648                                            660.     26.46
  649                                            720.     28.51
  650                                            780.     29.61

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+........
  651                                            840.     29.43
  652                                            900.     29.27
  653                                            960.     29.66
  654                                           1020.     29.64
  655  LAST DUMMY SURFACE TEMP (NEGATIVE)       - 10.
  656   DIFFUSIVITIES 1         360.0                 9.0       9.0     235.8
  657                           326.5     126.2     126.2
  658   DIFFUSIVITIES 2         420.0                41.1      41.1     165.8
  659                           140.1      26.6      26.6
  660   DIFFUSIVITIES 3         480.0               486.9     486.9     388.8

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+........
  661                           841.9     507.3     507.3
  662   DIFFUSIVITIES 4         540.0              1477.0    1477.0     536.8
  663                           866.3     526.5     526.5
  664   DIFFUSIVITIES 5         600.0              1781.4    1781.4    6200.8
  665                          9842.7    1766.4    1766.4
  666   DIFFUSIVITIES 6         660.0               137.9     137.9     738.8
  667                          1495.3     363.1     363.1
  668   DIFFUSIVITIES 7         720.0              3099.0    3099.0   11193.6
  669                         18901.8    6755.4    6755.4
  670   DIFFUSIVITIES 8         780.0              2136.2    2136.2    7653.7

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+........
  671                         13582.4   10763.3   10763.3
  672   DIFFUSIVITIES 9         840.0              1711.2    1711.2    7410.8
  673                         13552.5   10854.4   10854.4
  674   DIFFUSIVITIES10         900.0              1416.5    1416.5    5997.4
  675                         13092.3   10930.4   10930.4
  676   DIFFUSIVITIES11         960.0               103.1     103.1    5351.8
  677                          7507.8    2519.1    2519.1
  678   DIFFUSIVITIES12        1020.0               103.1     103.1    5352.0
  679                          7508.0    2519.2    2519.2
  680  LAST                     -10.0

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+........
  681  LAST                     -10.0
  682  END
```

Figure 5-1 (Continued)

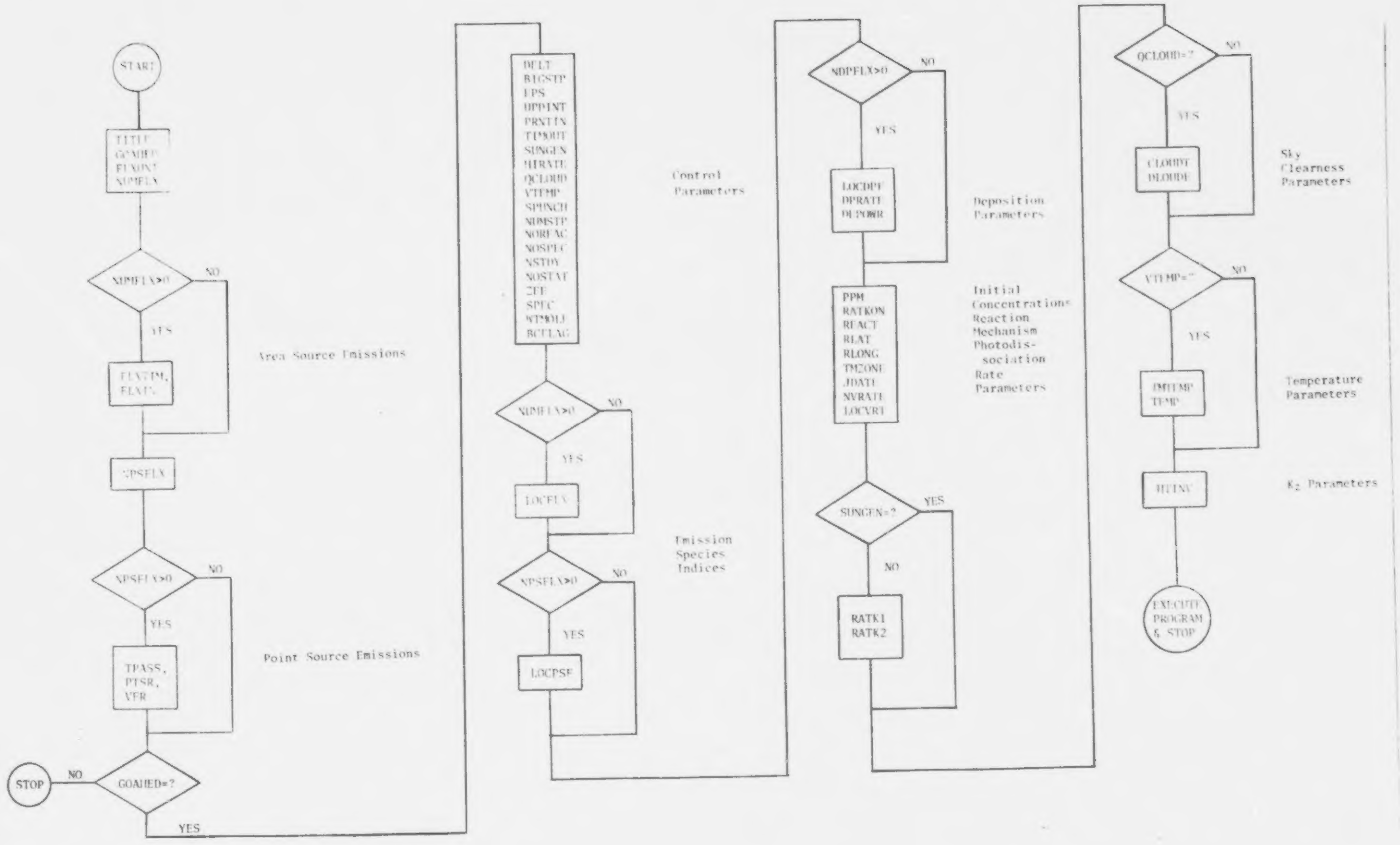


Figure 5-2 KEMOD Input Variable Sequence and Options

The program's inputs begin with the 80-character page label (TITLE) similar to that used in the other modules of the code. The page label is followed by a parameter (GOAHED), which indicates whether or not the program should proceed beyond the PREDAT listing of the input deck and reading of the emissions data schedule.

5.1.1 Source Emissions Data

The first card in the block of emissions data parameters specifies whether the emission data is input on the mass fraction or mole fraction basis. The user should specify either the word MOLE or MASS starting in column 41. When the emissions data are input on the mass fraction basis, the program converts it to a mole (volume) basis, using the molecular weights specified later in the input stream.

The next card read indicates the number of species with area source emissions data inputs. The user should specify a value of seven (7) corresponding to the species or species groups listed below:

1) Nitrogen oxides
2) Alkanes (paraffinic hydrocarbons)
3) Alkenes (olefinic hydrocarbons)
4) Aromatic hydrocarbons
5) Aldehyde hydrocarbons
6) Carbon monoxide
7) Sulfur oxides

Next, the area source emission fluxes are specified using a pair of cards for each entry in the schedule. Each pair contains a time (0-1440 minute clock) and seven species emission fluxes in the units of mole or mass fraction meters per minute. In the St. Louis application, the order of species' fluxes on the cards is indicated in the list shown above. In general, the order of the species' fluxes is arbitrary, subject to the constraint that nitrogen oxides (NO_x) and aldehydes must be first and fifth in the list. This constraint is necessary because the program internally partitions these emissions data to more than one species. More specifically, the program assumes the NO_x emissions are

90% nitric oxide (NO) and 10% nitrogen dioxide (NO_2). It assumes the aldehyde emissions are 60% formaldehyde (HCHO) and 40% other aldehyde species (RCHO). The list of area source emissions flux cards is terminated with a pair of cards, the first of which has a negative entry in columns 21-30. It is important to note that the program uses the initial and final times in the area source emissions schedule as the start time and stop time of the simulation.

Following the area source inputs, the program reads the point source emissions inputs. The number of species with point source emissions data (NPSFLX) is read first. The number of species here is normally seven, but may be less in situations where the user includes all hydrocarbon and/or carbon monoxide emissions sources as area sources. This card is followed by a temporal schedule, which normally (i.e., when NPSFLX>4) has three cards per point source. The first pair of cards indicates the time and mole or mass fraction-meters of species emissions entrained into the air parcel moving along the trajectory. (See Section 4 for an explanation of these units). The order in which the species emissions are listed on the cards is subject to the same constraints discussed above for area source emissions species. The third card of the set contains the normalized vertical distribution of the emissions with respect to the vertical mesh points. It is important to point out that the sum of the elements of this vertical distribution must equal one for proper representation of the source emissions. The point source data schedule is terminated by a pair of cards, the first of which has a negative entry in column 21-30.

5.1.2 Control Parameters and the Vertical Mesh

The program next reads a series of control parameters. The first card contains the initial integration time step size. This parameter is specified in minutes and must be quite small (10^{-3} to 10^{-5}) for proper start-up of the integration algorithm. Next, the program reads a card with the maximum integration time step size in minutes. The program's integration method is a variable step method, which chooses the time step at each point in the solution on the basis of certain convergence criteria. In practice, the maximum step size permitted is a necessary input to this algorithm, and values of 4 to 8 minutes are recommended.

Next, the user specifies the integration error control criteria. This is a dimensionless and somewhat empirical parameter, which should be input in the range of 10^{-3} to 10^{-4}. The program then reads a card which contains the update time interval in minutes. This is the interval at which the program checks for new values in the various schedules of time-varying input parameters. The update interval is used to revise the schedules of several inputs for use by the program, as described in Section 5.5. Values between 10 and 20 minutes are recommended.

The next card in the input deck specifies the time interval between printouts of the pollutant concentrations. This print interval must be chosen to be a multiple of the update interval for proper operation of the code.

The program next reads a series of cards with YES or NO inputs for various program options. The first card here indicates whether or not the program should print six integration progress parameters at each output interval. These parameters are discussed in Section 5.4. The next card indicates whether the user wants the program to generate the schedules of photodissociation rates (YES), or whether the user wants the data read in directly (NO). The following card also pertains to the photodissociation rates, and it indicates whether (YES), or not (NO), they vary with elevation. These last two cards should be input as YES under almost all conditions. See Section 5.3 for elucidation of the photodissociation rates used by the program. Next, the program reads an option flag, which indicates whether (YES), or not (NO), sky clearness ratios are included in the input deck. The program assumes clear sky conditions when they are absent. Continuing, the program reads a card which specifies whether (YES or NO) a schedule of temperatures is input. The program assumes an atmospheric temperature of $25^{\circ}C$ when this input is NO. The final YES or NO input indicates whether or not the user wants the computed ground concentrations punched at each print interval. This option is useful if the modeler wants to save the output in a form convenient for subsequent processing.

The program next reads a series of integer constants. The first is the maximum number of integration time steps permitted in a run. This number should be selected sufficiently large (i.e., $N \geq 75$ steps per hour of real-time simulation) so that termination of a run on the basis of this criterion occurs only in abnormal/erroneous conditions. Next, the user specifies the number of chemical reactions (NOREAC) and number of chemical species (NOSPEC) in the chemical mechanism. For the ERT photochemical mechanism, these parameters have values of 51 and 30, respectively. In addition, the user must specify the number of species (a subset of NOSPEC) which are treated as constants or treated as steady-state species in the model. There is only one such species, H_2O, in the ERT photochemical mechanism. The last integer constant read in the section is the number of vertical mesh points (NOSTAT) input. This card is followed by NOSTAT cards with the mesh point elevations specified in order of ascending elevations, as done in the input decks for other program modules.

5.1.3 Chemical Species List

The program reads a series of cards which each contain the name (up to 4 characters) of one chemical species, its molecular weight, and a code for the type of surface boundary condition it has. The species' names must be input in the order shown in Figure 5-1 for the ERT photochemical mechanism. The molecular weight data is completely optional when emissions data are specified in moles. When the emissions data is input on the mass fraction basis, only the molecular weights of the species with emissions data need be specified. The boundary condition code is input with an integer value of one when the user wishes to include surface deposition for that species. Otherwise, the user may input a zero or leave the field blank to indicate the default (no deposition) surface boundary condition.

5.1.4 Emission Species and Surface Deposition Species

The user inputs a set of cards which indicates the correspondence between area source emissions species and the species number (index) in the chemical species list. The first card indicates the chemical species index for the first entry on the area source emissions data card pair. In the sample problem, the first entry of emissions data is for NO_x; so the corresponding species index is 1 for NO. Similarly, for the second card here, the species index is input as 9 (for PA) since the alkane emissions are specified second on the emissions data card pairs, etc. A similar list of species indices is input for the point source emissions species, following the cards for the area source species indices.

Next, the program reads a set of cards with data for the species with surface deposition boundary conditions. The program reads one card for each species that has a boundary condition code equal to one. Each card contains one of the depositing species' index (from the species list), its surface deposition velocity (v) in meters/minute, and the exponent (p) of the concentration (c) in the analytic deposition flux (Φ_d) expression shown below:

$$\Phi_d = v*c^p$$

This exponent has a value of one for dry deposition, but may take on other values when modeling higher order surface chemical reactions.

5.1.5 Initial Concentrations

Initial pollutant concentrations are input using one card per species and with the species ordered as shown in the species list. Each card contains the initial concentrations at the vertical mesh points in the units of parts per million. Ground-level concentration is entered in columns 21-30; the concentration at the second mesh point is in columns 31-40; etc.

The reader can observe from Figure 5-1 that some cards are labeled COMPUTED and their concentration input fields are blank. Initial concentrations for these species are computed internally using the psuedo-steady-state assumption. Table 5-2 lists the species which require initial concentration input data and those which are computed. Note that one card for every species in the species list must be included, even though some cards have blank input fields.

5.1.6 Chemical Reaction Mechanism Inputs

The chemical reaction rate constants are specified next in the input deck. One rate constant per card is input in columns 41-50, and a total of NOREAC (See Section 5.1.2) cards must be input. The units of the rate constants are ppm^{-1} - $minute^{-1}$ for bimolecular reactions. The rate constants for ERT reaction numbers 1, 6, 21, 34, 37, and 38 may be left blank since they are either photodissociation rates or temperature

TABLE 5-2

USER SPECIFIED AND INTERNALLY COMPUTED INITIAL CONCENTRATIONS

User Specified	Internally Computed
NO	NO2
O3	HONO
CO	NO3
HCHO	PAN
RCHO	ARCO
OLEF	RO2
PA	RCO3
HO2	AO2
AR	PAO2
OH	D
SO2	HNO4
SO4	ARO
H2O	AROH
	AO
	O
	PAO
	N2O5

dependent rates which are determined elsewhere in the model.

The program next reads a block of cards (total number equals NOREAC) which describe the chemical mechanism. Each reaction is specified in alphanumeric format in columns 1-80. The information on these cards is used only to reproduce the mechanism in the output listing; it is not used in the computation. The stoichiometric coefficients for the ERT photochemical mechanism are contained in the program's explicit chemical rate expressions; thus, they are not a part of the input stream.

5.1.7 Photodissociation Rate Input Parameters

The program internally stores photodissociation rate data as functions of solar zenith angle and elevation above the surface. The user is required to specify the latitude, longitude, time zone, and date, so that the program can determine a temporal schedule of solar zenith angles and, thus, photodissociation rates. The longitude and latitude of the approximate center of the modeling region are first specified in degrees. Next, the time zone is indicated by the number of hours difference between Greenwich, England 0° meridian time and local standard time, followed by the date in the year-month-day form (YRMODA).

Next, the program reads the number and reaction number(s) of the independent photodissociation reactions. The program has built-in photodissociation rate data for NO_2 and HCHO photolysis, the two independent photodissociation rates. The program has been equipped to determine the photolysis rates for other species (HONO and RCHO) on the basis of the independent rates data. Thus, the user should specify this data exactly as shown in Figure 5-1 for the ERT photochemical mechanism.

The program has a seldom-used option which allows the user to input photodissociation rate constants. The user should specify SUNGEN = NO in the control parameter section of the input, and input two arrays for the photodissociation rates of NO_2 and HCHO. When the user wants to input different rates for the different mesh point elevations, HIRATE = YES is specified. The program then reads a block of cards which each has NOSTAT (See Section 5.1.2) entries of photodissociation rates for one reaction in the fields 21-32, 33-44, 45-56, 57-68, 69-80. The first card in the block must correspond to the start time of the simulation, and subsequent cards must be input for times at the frequency of the update interval. The last card in the block is input with a negative entry in columns 21-32. A second block of cards is input for the photodissociation rates of the second reaction with the same format. If the user selects HIRATE = NO, only one entry per card is required in columns 21-32.

5.1.8 Sky Clearness, Temperature, and Diffusivity Schedules

The program reads a schedule of sky clearness ratios. Each card in the schedule contains a time in minutes and the ratio of actual ultraviolet radiation to the expected clear sky radiation for that time, date and location. The ratio may have values between zero and one, where one indicates clear sky conditions. These ratios are used by the program to adjust the photodissociation rates to represent the actual ultraviolet radiation intensity. The last card in the schedule is input with a negative entry in columns 41-50.

Next, the program reads a temporal schedule of temperatures. Each card contains the time in minutes and temperature in degrees Celcius. The temperatures specified here are used by the model to determine the rate constants of chemical reaction numbers 21 and 37. The last entry in the schedule contains a negative value in columns 41-50.

Lastly, the program reads the schedule of eddy diffusivity (K_z) coefficients. A pair of cards is input for each hour which contains a time in minutes and NOSTAT +1 K_z coefficients. The first card contains the time in columns 21-30 and three K_z coefficients in columns 41-50, 51-60, and 61-70. The second card contains the remaining K_z coefficients starting in column 21. The coefficients on the cards are ordered by ascending elevations. As described in Section 3, these K_z coefficients are the average K_z coefficients between vertical mesh point elevations. The first and last K_z coefficients on the card pair are assigned values equal to the next innermost K_z value for use by the program. The reading of this schedule is terminated when the program reads a pair of cards with the entry for time set less than zero.

Each of the schedules described here are output by the meteorological module in the proper format. Nevertheless, it is important for the user to know that the program requires each of these schedules to start at the same time as the simulation start time. The schedules should extend to a time when the input parameter becomes invariant. For example, the user may find it convenient to end the sky clearness ratio schedule prior to the stop time if the sky is perfectly clear after a certain time of day. However, it is unlikely that this would be the case for temperature or K_z coefficients.

5.2 ERT Photochemical Mechanism

The model's chemical mechanism, as shown in Table 5-3, simulates the reactions occurring in an irradiated hydrocarbon/NO_x/SO_x mixture in air. The hydrocarbon reactivity is delineated by the use of five classes of hydrocarbon species in the mechanism. The primary purpose of the mechanism is to provide the chemical rates which control the formation of ozone, nitrogen dioxide, and sulfates in the atmosphere. This mechanism was developed by ERT and systematically tested against smog chamber data from the University of California, Riverside, chamber facility. The reader is referred to Table 5-4, which defines the chemical species symbols used in the mechanism.

The hydrocarbon reaction rate constants used in atmospheric application of the chemical mechanism are approximations based on the estimated reactivity of typical atmospheric hydrocarbon mixtures. The reactivity of the atmospheric alkanes is assumed to be somewhat greater than that of pure propane, but less than n-butane. The atmospheric alkenes are assumed to be as reactive as propene. The atmospheric aromatics have been assigned a reactivity greater than toluene but less than m-xylene. These rate constants may require adjustment for regions where the typical atmospheric hydrocarbon mixture is more or less reactive than St. Louis.

5.3 Photodissociation Rates

The ERT photochemical mechanism has four reactions which involve photolysis by ultraviolet radiation. These reactions play a particularly important role in the chemical processes which control ozone formation. In fact, without ultraviolet radiation, the rate of ozone formation in the troposhere would be negligible. They are also important as a source of the free radical species (OH, HO_2, and RO_2) that influence the HC/NO_x mechanism and the rate of sulfate formation.

The photlysis steps in the mechanism are for nitrogen dioxide, nitric acid, formaldehyde, and other aldehydes. The clear sky photodissociation rates for nitrogen dioxide and formaldehyde, as functions of

TABLE 5-3

THE ERT PHOTOCHEMICAL REACTION MECHANISM

		Rate Constants ($ppm^{-1}\ min^{-1}$) at 1 Atmosphere
1	$NO_2 + h\nu = O + NO$	Ultraviolet dependent
2	$O + O_2\ (+ M) = O_3\ (+ M)$	4.12E+06
3	$O_3 + NO = NO_2 + O_2$	2.50E+01
4	$NO + NO_2 + H_2O = 2HONO$	2.20E-09
5	$HONO + HONO = NO + NO_2 + H_2O$	1.40E-03
6	$HONO + h\nu = OH + NO$	Ultraviolet dependent
7	$OH + NO + M = HONO + M$	1.50E+04
8	$OH + NO_2 + M = HNO_3 + M$	1.50E+04
9	$OH + CO = HO_2 + CO_2$	4.40E+02
10	$HO_2 + NO = NO_2 + OH$	1.20E+04
11	$HO_2 + NO_2 = HNO_4$	1.70E+03
12	$HO_2 + HO_2 = H_2O_2 + O_2$	8.40E+03
13	$NO_2 + O_3 = NO_3 + O_2$	5.00E-02
14	$NO_3 + NO = 2NO_2$	2.80E+04
15	$NO_3 + NO_2 = N_2O_5$	9.30E+02
16	$N_2O_5 + H_2O = 2HNO_3$	5.00E-05
17	$N_2O_5 = NO_3 + NO_2$	2.40E+01
18	$OH + OLEF = AO_2$	3.70E+04
19	$AO_2 + NO = NO_2 + AO$	2.90E+04
20	$AO + O_2 = RCHO + HCHO + HO_2$	4.10E+05
21	$HNO_4 = HO_2 + NO_2$	Temperature dependent

TABLE 5-3 (CONTINUED)

		Rate Constants (ppm^{-1} min^{-1}) at 1 Atmosphere
22	O_3 + OLEF = 0.5HCHO + 0.5RCHO + 0.25HO_2 + 0.25RCO_3 + 0.5OH + 0.5$RCHO_2$	1.90E-02
23	$RCHO_2$ + NO_2 = NO_3 + 0.5HCHO + 0.5RCHO	2.30E+04
24	O + OLEF = .3EPOX + .3RCHO + .4HO_2 + .4RO_2	5.20E+03
25	$RCHO_2$ + NO = NO_2 + 0.5HCHO + 0.5RCHO	2.90E+04
26	OH + PA = H_2O + PAO_2	3.80E+03
27	PAO_2 + NO = NO_2 + .85PAO + .15RO_2	2.90E+04
28	PAO_2 + NO = NTRA	2.60E+03
29	PAO = RO_2 + .5HCHO + .5RCHO	1.40E+05
30	RO_2 + NO = NO_2 + PAO	2.90E+04
31	PAO + O_2 = .5KET + .5RCHO + HO_2	6.70E+04
32	PAO + NO_2 = .85NTRA + .15RCHO + .15HONO	2.30E+04
33	OH + RCHO = RCO_3 + H_2O	2.20E+04
34	RCHO + hν = RO_2 + HO_2 + CO	Ultraviolet dependent
35	RCO_3 + NO = CO_2 + NO_2 + RO_2	2.90E+04
36	RCO_3 + NO_2 = PAN	1.70E+04
37	PAN = RCO_3 + NO_2	Temperature dependent
38	HCHO + hν = .67HO_2 + CO	Ultraviolet dependent
39	HCHO + OH = H_2O + HO_2 + CO	2.10E+04
40	RO_2 + HO_2 = RO_2H + O_2	4.20E+03
41	AR + OH = AROH + HO_2	2.24E+04
42	AROH + OH = ARCO	5.02E+04

TABLE 5-3 (CONTINUED)

		Rate Constants (ppm^{-1} min^{-1}) at 1 Atmosphere
43	ARCO + NO = NO_2 + RCHO	2.90E+04
44	AR + OH = H_2O + ARO	5.60E+03
45	ARO + NO = NO_2 + HO_2 + $AR'CHO$	2.90E+04
46	SO_2 + O = SO_4	8.40E+00
47	SO_2 + OH = SO_4	1.76E+03
48	SO_2 + HO_2 = SO_4 + OH	3.00E-02
49	SO_2 + D = SO_4 + RCHO	2.90E+03
50	SO_2 + RO_2 = SO_4 + PAO	8.00E+00
51	SO_2 + AO_2 = SO_4 + AO	8.00E+00

TABLE 5-4

CHEMICAL SPECIES SYMBOL DEFINITIONS

Species	Symbol Designation
AO	Alkoxy radical equivalent to AO_2
AO_2	Product of OH addition to olefin in the presence of O_2
AR	Aromatic hydrocarbons
AR'CHO	Aromatic aldehyde
ARCO	Product of addition of OH to a cresol in the presence of O_2
ARO	Product of H-abstraction from side chain alkyl group on benzene ring followed by addition of O_2 to radical formed
AROH	Cresols
CO	Carbon monoxide
CO_2*	Carbon dioxide
D	Criegee intermediate ($RCHO_2$)
EPOX*	Epoxide formed from O atom addition to olefin
HCHO	Formaldehyde
HONO	Nitrous acid
HNO_3*	Nitric acid
HNO_4	Pernitric acid
HO_2	Hydroperoxyl radical
H_2O	Water
H_2O_2*	Hydrogen peroxide
HV	Photon
KET*	Ketones
M*	Any third body, such as N_2 or O_2
NO	Nitric oxide

TABLE 5-4 (CONTINUED)

Species	Symbol Designation
NO_2	Nitrogen dioxide
NO_3	Nitrate radical
N_2O_5	Dinitrogen pentoxide
NTRA*	Organic nitrate
O	Oxygen atom (ground state)
O_2*	Oxygen
O_3	Ozone
OH	Hydroxyl radical
OLEF	Alkenes (olefinic hydrocarbons)
PA	Alkanes (paraffinic hydrocarbons)
PAN	Peroxyacetyl Nitrate
PAO	Alkoxy radical formed by PA
PAO_2	Alkyl peroxy radical from the O_2 addition to the radical formed by H-abstraction from a paraffinic hydrocarbon
R**	Generalized alkyl group (e.g., C_2H_5, C_3H_7, etc.)
RCHO	Aldehydes other than formaldehyde
RCO_3	Acyl peroxy radical
RO	Alkoxyl radical
RO_2	Alkyl peroxy radical
RO_2H*	Product of disproportionation between HO_2 and RO_2
SO_2	Sulfur dioxide
SO_4	Sulfate

---

* Species' Concentration not computed by the model

**Definition of species group

elevation and solar zenith angle, are presented in Tables 5-5 and 5-6. The rates shown are 95% of the values reported by Peterson, 1977 and Peterson, et al 1976.

5.4 Description of the Output

This section describes the printed output of the KEMOD program. The output of the program includes the PREDAT listing of the input deck, partial verification of the input data, the computed species concentrations, and a printer-plot of the concentration history for five species. There is also an optional listing to verify the schedules of all time-varying parameters used by the program.

The first page of output following the PREDAT listing (see Figure 5-1) is a partial verification of the input control parameters. As shown in Figure 5-3, the program lists the initial and final simulation times, the initial and maximum integration time step sizes, and the number of reactions, species, steady-state species, and vertical mesh points.

The next page of output, Figure 5-4, lists area source emissions data for the seven emission species. The columns labeled NO and HCHO represent the total nitrogen oxides and aldehydes emissions fluxes, respectively. The user should note that the data has been rescheduled such that each entry is at a fixed time interval, the update interval described in Section 5.5. Also, the fluxes have been converted to the units of ppm-meters per minute.

Next, the program lists the sums of the effective point source emission rates by time, species, and vertical mesh points (or cells) as shown in Figure 5-5. For each time shown, the program lists NOSTAT lines of emission rates in ppm per minute. The first line refers to the emissions in the ground cell by species, the second line corresponds to the second cell, etc. The program converts the input data to the emission rates shown, as described in Section 5.5.

The next page of output is a listing of the ERT photochemical mechanism and the rate constants used for each of the reactions (See Figure 5-6). Note that the listing includes the initial photodissociation and temperature-dependent reaction rates computed internally. This page is followed by one with the chemical species list, as shown in Figure 5-7.

TABLE 5-5

NO_2 Photodissociation Rates (per minute)

$NO_2 + hv \xrightarrow{k} NO + O$

Solar Zenith Angle

Elevation Above Surface (meters)	0°	10°	20°	30°	40°	50°	60°	70°	78°	86°
0	.550	.545	.532	.508	.471	.416	.334	.219	.108	.024
150.	.583	.578	.566	.543	.507	.453	.371	.251	.126	.026
360.	.613	.608	.597	.574	.540	.486	.405	.280	.145	.028
640.	.641	.637	.626	.605	.571	.518	.436	.309	.165	.032
980.	.668	.665	.653	.634	.599	.547	.466	.335	.184	.035
1380.	.693	.689	.679	.659	.626	.574	.491	.359	.201	.038
1840.	.714	.711	.691	.681	.649	.597	.514	.380	.218	.041
2350.	.733	.730	.720	.701	.669	.617	.536	.400	.234	.044
2910.	.751	.748	.738	.720	.689	.639	.554	.421	.251	.047
3530.	.768	.765	.755	.737	.708	.659	.578	.443	.269	.052
4210.	.783	.780	.771	.754	.726	.677	.599	.465	.288	.057

TABLE 5-6

HCHO Photodissociation Rates* (per minute)

$$\mathrm{HCHO} + hv \xrightarrow{k} \begin{cases} .67\mathrm{HO_2} + .33\mathrm{CO} \\ .67\mathrm{CO} + \mathrm{HCO} \end{cases}$$

	Solar Zenith Angles									
Elevations Above Surface (meters)	0°	10°	20°	30°	40°	50°	60°	70°	78°	86°
0.	.621	.613	.584	.536	.467	.376	.266	.144	.060	.011
150.	.671	.662	.632	.583	.511	.414	.298	.164	.068	.013
360.	.719	.710	.680	.629	.555	.455	.329	.183	.077	.014
640.	.768	.758	.728	.676	.599	.494	.361	.204	.086	.015
980.	.817	.807	.776	.721	.642	.533	.393	.225	.095	.016
1380.	.858	.849	.816	.761	.679	.566	.422	.244	.104	.018
1840.	.907	.897	.864	.807	.723	.605	.454	.266	.114	.019
2350.	.943	.933	.899	.843	.757	.638	.482	.286	.124	.020
2910.	.983	.972	.939	.882	.795	.672	.512	.308	.136	.022
3530.	.983	.972	.939	.882	.795	.672	.512	.308	.136	.022
4210.	.983	.972	.939	.882	.795	.672	.512	.308	.136	.022

*All values shown have the exponent 10^{-2}.

```
RT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS -  6/29/76   (181)

NPUT DATA FOR VERTICAL DIFFUSION WITH PHOTOCHEMISTRY

    INITIAL TIME IS                      360.00 MINUTES

    FINAL TIME IS                       1020.00 MINUTES

    INITIAL TIME STEP SIZE IS          .000100 MINUTES

    MAXIMUM TIME STEP SIZE IS             6.00 MINUTES

    NUMBER OF REACTIONS IS                  51

    NUMBER OF SPECIES IS                    30

    NUMBER OF SPECIES IN STEADY STATE IS     1

    NUMBER OF VERTICAL MESH POINTS IS        5    INCLUDING THE GROUND AND THE EDGE
```

Figure 5-3

```
ERT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS -  6/29/76   (181)

REVISED AREA SOURCE FLUX SCHEDULE          ---        PPM-METERS/MINUTE

TIME        NO           PA           OLEF          AR           HCHO          CO           SO2

360.0     0.           0.           0.           0.           0.           0.           0.
375.0     0.           0.           0.           0.           0.           0.           0.
390.0      .1301E-03    .9809E-05    .6396E-05    .4467E-05    .3148E-05    .8256E-03    .7077E-0
405.0      .6295E-03    .4746E-04    .3095E-04    .2161E-04    .1523E-04    .3995E-02    .3424E-0
420.0      .8311E-03    .1606E-03    .1088E-03    .5272E-04    .3435E-04    .9470E-02    .4057E-0
435.0      .1399E-02    .3837E-03    .2840E-03    .1358E-03    .8470E-04    .2371E-01    .6006E-0
450.0      .1399E-02    .3837E-03    .2840E-03    .1358E-03    .8470E-04    .2371E-01    .6006E-04
465.0      .1399E-02    .3837E-03    .2840E-03    .1358E-03    .8470E-04    .2371E-01    .6006E-04
480.0      .1327E-02    .3756E-03    .2687E-03    .1267E-03    .7899E-04    .2190E-01    .5760E-04
495.0      .9742E-03    .3807E-03    .2011E-03    .7500E-04    .4506E-04    .1196E-01    .4470E-04
510.0      .6356E-03    .1597E-03    .7141E-04    .3208E-04    .1847E-04    .4525E-02    .3293E-04
525.0      .5345E-03    .1116E-03    .5377E-04    .2721E-04    .1632E-04    .4176E-02    .2836E-04
540.0      .4313E-03    .1224E-03    .5266E-04    .2639E-04    .1359E-04    .3346E-02    .2246E-04
555.0      .7403E-03    .4262E-03    .1980E-03    .7363E-04    .4702E-04    .6062E-02    .4079E-04
570.0      .4319E-03    .1211E-03    .3691E-04    .5692E-04    .1741E-04    .3386E-02    .1836E-04
585.0      .1992E-02    .2333E-02    .1009E-02    .3830E-03    .1634E-03    .4185E-01    .5007E-04
600.0      .5200E-02    .5881E-02    .3727E-02    .1725E-02    .1127E-02    .1974E+00    .1858E-03
615.0      .8387E-02    .5010E-02    .3380E-02    .1445E-02    .8860E-03    .2507E+00    .2681E-03
630.0      .1780E-01    .9854E-02    .7877E-02    .3173E-02    .2302E-02    .7371E+00    .4582E-03
645.0      .3793E-01    .2061E-01    .1473E-01    .6640E-02    .4405E-02    .1046E+01    .1108E-02
660.0      .6737E-01    .3404E-01    .2401E-01    .1034E-01    .6565E-02    .2048E+01    .3245E-02
675.0      .9343E-01    .3515E-01    .2449E-01    .1184E-01    .7056E-02    .2141E+01    .4570E-02
690.0      .8570E-01    .3253E-01    .2184E-01    .1040E-01    .6053E-02    .1574E+01    .2466E-02
705.0      .3136E+00    .7795E-01    .5353E-01    .2441E-01    .1572E-01    .4309E+01    .1204E-01
720.0      .5842E-01    .1451E-01    .9854E-02    .4694E-02    .3881E-02    .6278E+00    .3545E-02
735.0      .7291E-01    .4343E-01    .2419E-01    .8197E-02    .5368E-02    .1700E+01    .2901E-02
750.0      .4734E-01    .1602E-01    .1128E-01    .4181E-02    .3656E-02    .8330E+00    .3077E-02
765.0      .2359E-01    .8004E-02    .5766E-02    .2070E-02    .1558E-02    .4178E+00    .6708E-03
780.0      .1287E-01    .4408E-02    .3420E-02    .1306E-02    .1033E-02    .2903E+00    .4639E-03
795.0      .1289E-01    .5863E-02    .4263E-02    .1570E-02    .1181E-02    .3565E+00    .4793E-03
810.0      .8590E-02    .2317E-02    .1624E-02    .6246E-03    .4658E-03    .1167E+00    .3038E-03
825.0      .1098E-01    .4351E-02    .2582E-02    .7891E-03    .5514E-03    .1375E+00    .3245E-03
840.0      .1042E-01    .2187E-02    .1651E-02    .6628E-03    .4860E-03    .1162E+00    .3310E-03
855.0      .6665E-02    .1545E-02    .1098E-02    .4268E-03    .3111E-03    .7907E-01    .2540E-03
870.0      .1841E-02    .4134E-03    .3057E-03    .1238E-03    .9088E-04    .2357E-01    .7492E-04
885.0     0.           0.           0.           0.           0.           0.           0.
900.0     0.           0.           0.           0.           0.           0.           0.
915.0     0.           0.           0.           0.           0.           0.           0.
930.0     0.           0.           0.           0.           0.           0.           0.
945.0     0.           0.           0.           0.           0.           0.           0.
960.0     0.           0.           0.           0.           0.           0.           0.
975.0     0.           0.           0.           0.           0.           0.           0.
990.0     0.           0.           0.           0.           0.           0.           0.
1005.0    0.           0.           0.           0.           0.           0.           0.
1020.0    0.           0.           0.           0.           0.           0.           0.
```

Figure 5-4

```
-RT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS -  6/29/76   (181)

  POINT SOURCE EMISSION RATES BY VERTICAL CELL                    (PPM/MIN)

  TIME            NO          PA          OLEF         AR          HCHO         CO          SO2

 750.0        .140E-03    .149E-04    .519E-05    .868E-04    .201E-06    .166E-02    .345E-03
              .745E-04    .119E-04    .411E-05    .618E-04    .893E-07    .125E-02    .219E-03
              .925E-04    .831E-05    .294E-05    .433E-04    .130E-06    .864E-03    .195E-03
              .249E-04    .257E-05    .836E-06    .135E-04    .350E-07    .264E-03    .543E-04
             0.          0.          0.          0.          0.          0.          0.

 765.0        .116E-04    .292E-06    .143E-07    .134E-06    .970E-08    .779E-06    .321E-04
              .475E-05    .231E-06    .764E-08    .107E-06    .398E-08    .331E-06    .178E-04
              .729E-05    .161E-06    .862E-08    .739E-07    .612E-08    .488E-06    .196E-04
              .186E-05    .580E-07    .249E-08    .267E-07    .158E-08    .125E-06    .461E-05
             0.          0.          0.          0.          0.          0.          0.

 780.0        .857E-05    .345E-06    .318E-07    .133E-06    .718E-08    .418E-06    .197E-04
              .338E-05    .273E-06    .223E-07    .106E-06    .284E-08    .165E-06    .784E-05
              .544E-05    .189E-06    .179E-07    .734E-07    .456E-08    .265E-06    .125E-04
              .137E-05    .666E-07    .534E-08    .265E-07    .117E-08    .667E-07    .326E-05
             0.          0.          0.          0.          0.          0.          0.

 795.0        .169E-06    .553E-07    .206E-07    .156E-09    .468E-09    .186E-07    .115E-06
              .127E-06    .435E-07    .162E-07    .117E-09    .350E-09    .139E-07    .863E-07
              .855E-07    .299E-07    .111E-07    .787E-10    .236E-09    .936E-08    .582E-07
              .256E-07    .910E-08    .338E-08    .236E-10    .707E-10    .280E-08    .174E-07
             0.          0.          0.          0.          0.          0.          0.

 810.0        .169E-06    .468E-09    .468E-09    .156E-09    .468E-09    .186E-07    .115E-06
              .127E-06    .350E-09    .350E-09    .117E-09    .350E-09    .139E-07    .863E-07
              .855E-07    .236E-09    .236E-09    .787E-10    .236E-09    .936E-08    .582E-07
              .256E-07    .707E-10    .707E-10    .236E-10    .707E-10    .280E-08    .174E-07
             0.          0.          0.          0.          0.          0.          0.

 825.0        .773E-06    .813E-09    .813E-09    .271E-09    .813E-09    .363E-07    .762E-07
              .598E-06    .629E-09    .629E-09    .210E-09    .629E-09    .281E-07    .590E-07
              .396E-06    .416E-09    .416E-09    .139E-09    .416E-09    .186E-07    .390E-07
              .104E-06    .109E-09    .109E-09    .363E-10    .109E-09    .487E-08    .102E-07
             0.          0.          0.          0.          0.          0.          0.

 840.0        .773E-06    .813E-09    .813E-09    .271E-09    .813E-09    .363E-07    .762E-07
              .598E-06    .629E-09    .629E-09    .210E-09    .629E-09    .281E-07    .590E-07
              .396E-06    .416E-09    .416E-09    .139E-09    .416E-09    .186E-07    .390E-07
              .104E-06    .109E-09    .109E-09    .363E-10    .109E-09    .487E-08    .102E-07
             0.          0.          0.          0.          0.          0.          0.

 855.0       0.          0.          0.          0.          0.          0.          0.
             0.          0.          0.          0.          0.          0.          0.
             0.          0.          0.          0.          0.          0.          0.
             0.          0.          0.          0.          0.          0.          0.
             0.          0.          0.          0.          0.          0.          0.
```

Figure 5-5

The molecular weights and boundary condition codes are listed with each of the species numbers and names.

The next page of output, Figure 5-8, describes the vertical mesh geometry and the initial diffusion coefficients. Each vertical mesh point (or station in the listing) has its elevation above ground listed in meters. The initial eddy diffusivity coefficients are listed in a manner which shows that there is one K_z coefficient associated with the first station, one K_z coefficient between each pair of adjacent stations, and one K_z coefficient associated with the top (or edge) station. The vertical distance between the mesh points is printed under the column labeled DELTA Z. The figures in the last column indicate the cell height, i.e., the height of the volume of air associated with each cell.

The subsequent page of output lists the initial concentrations for all of the species at all the mesh point elevations, as shown in Figure 5-9. The list includes the concentrations of the species with internally computed initial concentrations.

The program then begins to output a series of pages at the frequency of the print interval which are similar to Figure 5-10. Each page contains the time in minutes (0-1440) and in hours (0-2400). A summary of several parameters for the surface cell is included. The values of the surface cell parameters shown are the last ones used before the times shown (updating of these and other parameters occurs after the printout is completed). The surface parameters listed are the NO_2 photodissociation rate, the eddy diffusivity coefficient, and the area and point source emission rates in ppm/minute. Note that the list includes emission rates for nine species, which is the result of the program allocating the NO_x emissions to NO and NO_2, and the aldehyde emissions to HCHO and RCHO. Next, the printout lists the species' names and the corresponding computed pollutant concentrations in parts per million at the mesh points heights. These, of course, are the most important results of the model.

Figure 5-10 also shows the optional output of integration progress parameters for the case with TIMOUT = YES. The listing includes the integration indexx which is normally equal to two. If the EPISODE algorithm encounters erroneous/abnormal conditions which prevent successful integration, the integration index is returned with a negative value and an error message describing the error. The listing shows the last

```
RT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS -  6/29/76    (181)
                                                                                              RATE CONSTANTS (1/(PPM--MIN))
                  R E A C T I O N S

  1.   1.NO2  +    HV               =  1.O    +  1.NO                                              .7041E-01
  2.   1.O    +    O2   +    M      =  1.O3   +    M                                               .4120E+07
  3.   1.O3   +  1.NO               =  1.NO2  +    O2                                              .2500E+02
  4.   1.NO   +  1.NO2  +  1.H2O    =  2.HONO                                                      .2200E-08
  5.   2.HONO                       =  1.NO   +  1.NO2  +  1.H2O                                   .1400E-02
  6.   1.HONO +    HV               =  1.OH   +  1.NO                                              .1971E-01
  7.   1.OH   +  1.NO   +    M      =  1.HONO +    M                                               .1500E+05
  8.   1.OH   +  1.NO2  +    M      =  1.HNO3 +    M                                               .1500E+05
  9.   1.OH   +  1.CO               =  1.HO2  +    CO2                                             .4400E+03
 10.   1.HO2  +  1.NO               =  1.NO2  +  1.OH                                              .1200E+05
 11.   1.HO2  +  1.NO2              =  1.HNO4                                                      .1700E+04
 12.   2.HO2                        =    H2O2 +    O2                                              .8400E+04
 13.   1.NO2  +  1.O3               =  1.NO3  +    O2                                              .5000E-01
 14.   1.NO3  +  1.NO               =  2.NO2                                                       .2800E+05
 15.   1.NO3  +  1.NO2              =  1.N2O5                                                      .9300E+03
 16.   1.N2O5 +  1.H2O              =  2.HNO3                                                      .5000E-04
 17.   1.N2O5                       =  1.NO3  +  1.NO2                                             .2400E+02
 18.   1.OH   +  1.OLEF             =  1.AO2                                                       .3700E+05
 19.   1.AO2  +  1.NO               =  1.NO2  +  1.AO                                              .2900E+05
 20.   1.AO   +    O2               =  1.RCHO +  1.HCHO +  1.HO2                                   .4100E+06
 21.   1.HNO4                       =  1.HO2  +  1.NO2                                             .3404E+01
 22.   1.O3   +  1.OLEF             =  .5HCHO +.5RCHO +.25HO2 +.25RCO3 +.5OH +.5O                  .1900E-01
 23.   1.O    +  1.NO2              =  1.NO3  +  .5HCHO +  .5RCHO                                  .2300E+05
 24.   1.O    +  1.OLEF             =  .3EPOX +  .3RCHO +  .4HO2  +  .4RO2                         .5200E+04
 25.   1.O    +  1.NO               =  1.NO2  +  .5HCHO +  .5RCHO                                  .2900E+05
 26.   1.OH   +  1.PA               =  1.H2O  +  1.PAO2                                            .3800E+04
 27.   1.PAO2 +  1.NO               =  1.NO2  + .85PAO  + .15RO2                                   .2900E+05
 28.   1.PAO2 +  1.NO               =    NTRA                                                      .2600E+04
 29.   1.PAO                        =  1.RO2  +  .5HCHO +  .5RCHO                                  .1400E+06
 30.   1.RO2  +  1.NO               =  1.NO2  +  1.PAO                                             .2900E+05
 31.   1.PAO  +    O2               =  .5KET  +  .5RCHO +  1.HO2                                   .6700E+05
 32.   1.PAO  +  1.NO2              = .85NTRA + .15RCHO + .15HONO                                  .2300E+05
 33.   1.OH   +  1.RCHO             =  1.RCO3 +  1.H2O                                             .2200E+05
 34.   1.RCHO +    HV               =  1.RO2  +  1.HO2  +  1.CO                                    .1377E-03
 35.   1.RCO3 +  1.NO               =  1.RO2   + 1.NO2  +   CO2                                   .2900E+05
 36.   1.RCO3 +  1.NO2              =  1.PAN                                                       .1700E+05
 37.   1.PAN                        =  1.RCO3 +  1.NO2                                             .1259E-02
 38.   1.HCHO +    HV               = .67HO2  +  1.CO                                              .4132E-03
 39.   1.HCHO +  1.OH               =  1.H2O  +  1.HO2  +  1.CO                                    .2100E+05
 40.   1.RO2  +  1.HO2              =    RO2H +    O2                                              .4200E+04
 41.   1.AR   +  1.OH               =  1.AROH +  1.HO2                                             .2240E+05
 42.   1.AROH +  1.OH               =  1.ARCO                                                      .5020E+05
 43.   1.ARCO +  1.NO               =  1.NO2  +  1.RCHO                                            .2900E+05
 44.   1.AR   +  1.OH               =  1.H2O  +  1.ARO                                             .5600E+04
 45.   1.ARO  +  1.NO               =  1.NO2  +  1.HO2  +    BALD                                  .2900E+05
 46.   1.SO2  +  1.O                =  1.SO4                                                       .8400E+01
 47.   1.SO2  +  1.OH               =  1.SO4                                                       .1760E+04
 48.   1.SO2  +  1.HO2              =  1.SO4  +  1.OH                                              .3000E-01
 49.   1.SO2  +  1.O                =  1.SO4  +  1.RCHO                                            .2900E+04
 50.   1.SO2  +  1.RO2              =  1.SO4  +  1.PAO                                             .8000E+01
 51.   1.SO2  +  1.AO2              =  1.SO4  +  1.AO                                              .8000E+01
```

Figure 5-6

```
ERT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS -  6/29/76   (181)

     SPECIES       MOLECULAR WEIGHT     BOUNDARY CONDITION

  1.  NO                  30.0                0
  2.  NO2                 46.0                0
  3.  O3                  48.0                1
  4.  HONO                47.0                0
  5.  CO                  28.0                0
  6.  HCHO                30.0                0
  7.  RCHO                58.0                0
  8.  OLEF                42.0                0
  9.  PA                  58.0                0
 10.  NO3                 62.0                0
 11.  HO2                 33.0                0
 12.  PAN                121.0                0
 13.  AR                  92.0                0
 14.  ARCO               156.0                0
 15.  RO2                 89.0                0
 16.  RCO3               103.0                0
 17.  AO2                 74.0                0
 18.  PAO2                89.0                0
 19.  D                   74.0                0
 20.  HNO4                79.0                0
 21.  ARO                123.0                0
 22.  AROH               107.0                0
 23.  AO                  58.0                0
 24.  OH                  17.0                0
 25.  O                   16.0                0
 26.  PAO                 73.0                0
 27.  N2O5               108.0                0
 28.  SO2                 64.0                1
 29.  SO4                 96.0                1
 30.  H2O                 18.0                0
```

Figure 5-7

```
ERT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS - 6/29/76   (181)     CP TIME =    0.00  DATE
          INITIAL DIFFUSION COEFFICIENTS
          VALUES SHOWN ARE FOR POINTS HALFWAY BETWEEN STATIONS

          STATION          HEIGHT (METERS)          DIFFUSIVITY               DELTA 7             CELL HEIGHT
             1                       0.00            9.0000000                                          60.00
                                                     9.0000000                 120.00
             2                     120.00                                                              150.00
                                                   235.80000                   180.00
             3                     300.00                                                              240.00
                                                   326.50000                   300.00
             4                     600.00                                                              450.00
                                                   126.20000                   600.00
             5                    1200.00          126.20000                                           300.00
```

Figure 5-8

```
ERT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS - 6/29/76   (181)     CP TIME =    0.00  DATE                PAGE   10

                                            TIME =  360.00 MINUTES     ( 600 ON 2400-CLOCK)

                                          INITIAL CONCENTRATION PROFILES IN PARTS PER MILLION



HEIGHT-M.        NO             NO2            O3             HONO           CO             HCHO           RCHO           OLEF
                 PA             NO3            HO2            PAN            AR             ARCO           RO2            RCO3
                 AO2            PAO2           O              HNO4           ARO            AROH           AO             OH
                 O              PAO            N2O5           SO2            SO4            H2O

   0.00       .30000E-02     .24501E-06     .29000E-01     .22827E-04     .13300E+00     .40000E-03     .20000E-03     .60000E-03
              .30000E-02     .43568E-11     .10000E-05     .48164E-08     .60000E-03     .15448E-08     .81679E-08     .14557E-08
              .25517E-08     .12025E-08     .16999E-08     .12237E-09     .38621E-09     .26773E-03     .54146E-12     .10000E-07
              .41869E-14     .38625E-11     .39950E-16     .32000E-02     .32000E-03     .17000E+05

 120.00       .25000E-02     .34134E-06     .43000E-01     .16872E-04     .11000E+00     .34000E-03     .17000E-03     .51000E-03
              .25400E-02     .10806E-10     .10000E-05     .90004E-08     .51000E-03     .15757E-08     .98732E-08     .19527E-08
              .26024E-08     .12218E-08     .28733E-08     .17048E-09     .39393E-09     .22757E-03     .46024E-12     .10000E-07
              .65767E-14     .38217E-11     .13804E-15     .28000E-03     .28000E-03     .17000E+05

 300.00       .15000E-02     .31254E-06     .58000E-01     .89473E-05     .66000E-01     .20000E-03     .10000E-03     .40000E-03
              .15000E-02     .22231E-10     .10000E-05     .10153E-07     .30000E-03     .15448E-08     .11331E-07     .24057E-08
              .25517E-08     .12025E-08     .37994E-08     .15610E-09     .38621E-09     .13386E-03     .27073E-12     .10000E-07
              .68131E-14     .25960E-11     .26003E-15     .16000E-02     .16000E-02     .17000E+05

 600.00       .15000E-02     .35912E-06     .58000E-01     .77869E-05     .66000E-01     .20000E-03     .10000E-03     .30000E-03
              .15000E-02     .25544E-10     .10000E-05     .11666E-07     .30000E-03     .15448E-08     .11495E-07     .24057E-08
              .25517E-08     .12025E-08     .37993E-08     .17936E-09     .38621E-09     .13386E-03     .27073E-12     .10000E-07
              .89950E-14     .26304E-11     .34330E-15     .16000E-02     .16000E-02     .17000E+05

1200.00       .15000E-02     .42311E-06     .58000E-01     .66092E-05     .66000E-01     .20000E-03     .10000E-03     .30000E-03
              .15000E-02     .30095E-10     .10000E-05     .13745E-07     .30000E-03     .15448E-08     .11743E-07     .24057E-08
              .25517E-08     .12025E-08     .37992E-08     .21133E-09     .38621E-09     .13386E-03     .27073E-12     .10000E-07
              .12486E-13     .26825E-11     .47655E-15     .16000E-02     .16000E-02     .17000E+05
```

Figure 5-9

```
ERT CHEMICAL-DIFFUSION MODULE SIMULATION   --   ST. LOUIS - 6/29/76   (181)

                              TIME =  780.00 MINUTES      (1300 ON 2400-CLOCK)

                              SURFACE CELL PARAMETERS:

                              NO2 PHOTODISSOCIATION RATE  =      .476   (/MIN)
                              DIFFUSIVITY COEFFICIENT     =   2376.9   (M**2/MIN)*

                              AREA SOURCES  (PPM/MIN)   POINT SOURCES  (PPM/MIN)

                                 NO    =     .193E-03          NO    =     .771E-05
                                 PA    =     .735E-04          PA    =     .345E-06
                                 OLEF  =     .570E-04          OLEF  =     .318E-07
                                 AR    =     .218E-04          AR    =     .133E-06
                                 HCHO  =     .103E-04          HCHO  =     .431E-08
                                 CO    =     .484E-02          CO    =     .418E-06
                                 SO2   =     .773E-05          SO2   =     .197E-04
                                 NO2   =     .215E-04          NO2   =     .857E-06
                                 RCHO  =     .689E-05          RCHO  =     .287E-08


                              CONCENTRATION PROFILES IN PARTS PER MILLION

 HEIGHT-M.    NO            NO2           O3            HONO          CO            HCHO          RCHO          OLEF
              PA            NO3           HO2           PAN           AR            ARCO          RO2           RCO3
              AO2           PAO2          O             HNO4          ARO           AROH          AO            OH
              O             PAO           N2O5          SO2           SO4           H2O

   0.00     .43947E-02    .10463E-01    .45896E-01    .29387E-04    .32687E+00    .12767E-02    .17996E-02    .26958E-02
            .65820E-02    .20050E-06    .66450E-06    .19683E-03    .21745E-02    .64978E-08    .13951E-06    .12369E-07
            .48761E-07    .11227E-07    .30302E-08    .15272E-05    .59559E-08    .26463E-03    .15164E-10    .62325E-07
            .12094E-08    .91666E-10    .78474E-07    .68282E-02    .13768E-02    .17000E+05

 120.00     .41444E-02    .98543E-02    .47092E-01    .29259E-04    .30995E+00    .12397E-02    .17761E-02    .24878E-02
            .62954E-02    .20530E-06    .72493E-06    .19740E-03    .20767E-02    .74607E-08    .15457E-06    .13623E-07
            .51714E-07    .12339E-07    .30356E-08    .15679E-05    .65366E-08    .26444E-03    .15166E-10    .67551E-07
            .11969E-08    .95733E-10    .75670E-07    .68289E-02    .13890E-02    .17000E+05

 300.00     .41026E-02    .94554E-02    .47706E-01    .29435E-04    .30202E+00    .12224E-02    .17664E-02    .23887E-02
            .61491E-02    .20156E-06    .76946E-06    .19803E-03    .20157E-02    .81218E-08    .16583E-06    .14653E-07
            .54113E-07    .13135E-07    .30410E-08    .15966E-05    .69142E-08    .26415E-03    .15710E-10    .72878E-07
            .12064E-08    .10162E-09    .71313E-07    .67400E-02    .13937E-02    .17000E+05

 600.00     .40700E-02    .90171E-02    .48303E-01    .29780E-04    .29425E+00    .12057E-02    .17575E-02    .22923E-02
            .59972E-02    .19617E-06    .82536E-06    .19895E-03    .19407E-02    .89387E-08    .17990E-06    .15996E-07
            .57312E-07    .14138E-07    .30545E-08    .16325E-05    .73468E-08    .26342E-03    .16506E-10    .79795E-07
            .12164E-08    .10932E-09    .66205E-07    .65348E-02    .13974E-02    .17000E+05

1200.00     .38435E-02    .81287E-02    .49390E-01    .30119E-04    .27652E+00    .11663E-02    .17328E-02    .20827E-02
            .56302E-02    .19145E-06    .93964E-06    .20147E-03    .17225E-02    .10755E-07    .20941E-06    .19029E-07
            .63882E-07    .16283E-07    .30993E-08    .16753E-05    .79995E-08    .25837E-03    .17374E-10    .92446E-07
            .11767E-08    .12010E-09    .58249E-07    .58212E-02    .14026E-02    .17000E+05

    INTEGRATION INDEX =    2          LAST STEP SIZE =     .141E+01                LAST ORDER USED =  2

    NUMBER OF STEPS =     220         NUMBER OF FUNCTION EVALUATIONS =  336        NUMBER OF JACOBIAN EVALUATIONS =  127
```

Figure 5-10

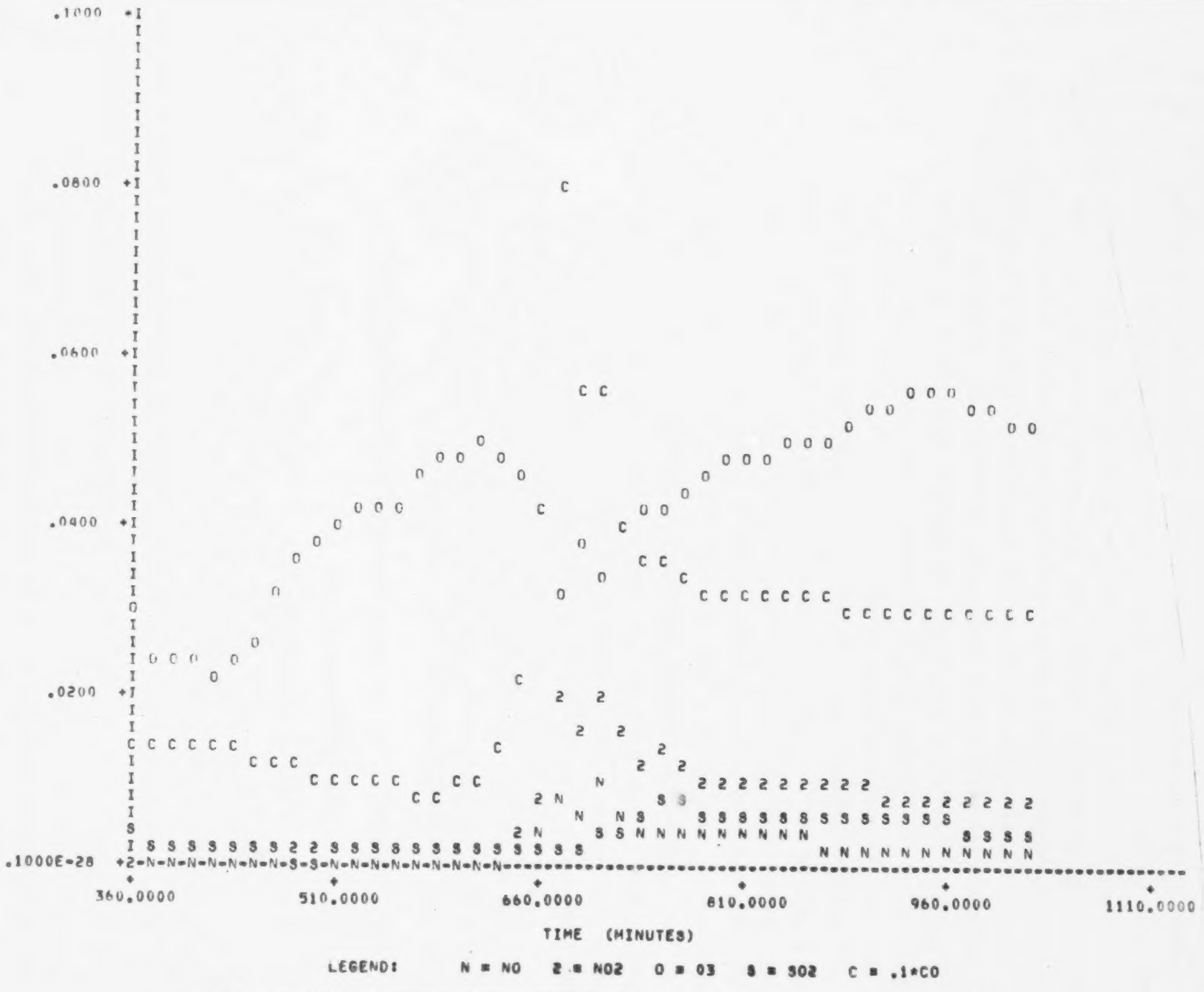
integration step size (in minutes) and the last order of the approximation used. The order refers to the number of previous time step concentrations and derivatives used to predict the concentrations at that time. The value of the order varies between one and five, depending on how rapidly the solution is varying. The listing shows the number of time steps, the number of function evaluations, and the number of Jacobian evaluations. Since the method used is a predictor-corrector method, the integration algorithm should always take more function evaluations than time steps. Similarly, the program is designed to minimize the number of Jacobian evaluations; thus, the number of Jacobians should always be less than the number of steps taken.

The last printed page of output from the chemical-diffusion model is a printer-plot of surface concentrations in ppm versus time. Figure 5-11 shows a plot of the concentration histories for the sample problem. The legend indicates the plotting symbols for the five species included in the plot. The nitric oxide, nitrogen dioxide, ozone, sulfur dioxide, and one-tenth carbon monoxide concentrations are plotted on the same page. Note that when two or more species have approximately the same concentrations at a plotting time, only the symbol associated with the last species in the legend with that value appears on the plot.

The user may select a seldom-used option of the KEMOD program which provides verification of the times and values the code has used in updating its parameters from the various temporal schedules. When this option is used, the following data is outputted.

- Photodissociation rates for all mesh point elevations
- Area source emission fluxes
- Point source emission rates
- Temperatures
- Sky clearness ratios
- K_z coefficients

The photodissociation rates, emission rates, and K_z coefficients are printed at the frequency of the update interval. The temperatures and sky clearness ratios are printed at the frequencies with which their values are changed. This results in a very sizable printout. This data



ERT CHEMICAL-DIFFUSION MODULE SIMULATION -- ST. LOUIS - 6/29/76 (CST)

Figure 5-11

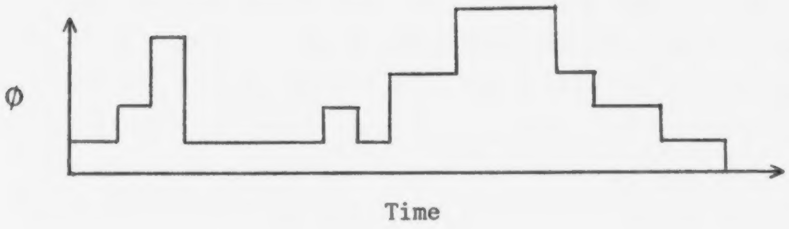


Figure 5-12 (a) Original Schedule

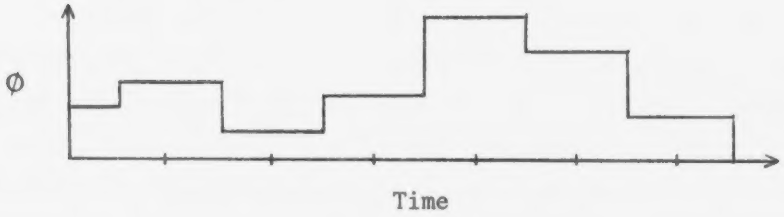


Figure 5-12 (b) Revised Schedule

is printed on logical unit numbers 10 and 11. Thus, to obtain the printout, the user must supply the appropriate installation-dependent job control such that these files are disposed to the line printer following the normal execution of the program.

5.5 Computational Procedures

Once the program has read the input data as described in Section 5.1, it performs various initialization tasks prior to passing control to the integration algorithm where the concentrations are computed. One common feature of most of the initialization tasks involves creating or revising the temporal schedules of parameters so that the entries in the schedules occur at a fixed time interval. In the subsequent discussion, we will refer to the program's 'update schedule', which is a schedule with a fixed time interval equal to the user-specified update interval.

The first initialization task involves generating the schedules of NO_2 and HCHO photodissociation rates for the duration of the run. The schedules are created by subroutine PHOTOD on the program's update schedule with entries for each mesh point elevation.

Next, the program updates the sky clearness ratio and determines the initial photodissociation rates in subroutines UPRAT2 and RATEHI. Also, the initial temperature-dependent rate constants are determined in subroutine TEMPR.

The program then proceeds to transform the area source emissions input data to the program's update schedule. The revision of the schedule involves an Euler integration of the emission rates to find the mean rate at each time in the new schedule. Figure 5-12 illustrates a typical input schedule and revised schedule of emission rates for one species. This transformation, of course, conserves the mass of emissions injected into the air parcel. Following the rescheduling (performed in subroutine SKEDUL), the program converts the area source emission fluxes to the units of ppm-meters/minute.

The next initialization task involves determining the point source emission rates from the input data. The program first truncates the time at which the source is passed by the air parcel to the nearest time in the program's update schedule. Second, it converts the mass or mole fraction-meters of emissions entrained into the air parcel to a rate assuming an entrainment/mixing period of twice the update interval. Third, it allocates the emission rates to the appropriate vertical mesh points, using the vertical distribution input for each source. The final result of this task is a schedule of the sums of point source emission rates by species and mesh point on the program's update schedule.

The next initialization task involves smoothing the K_z coefficient schedule. The program linearly interpolates the hourly K_z coefficients to determine intermediate values which coincide with the times in the program's update schedule. This task is performed to represent the mixing process as one which gradually changes with time, as opposed to one which may have order-of-magnitude changes in coefficients on the hour.

The last significant initialization task is the computation of the initial concentrations of species not specified in the input stream. The calculation is performed in subroutine ISTATE, which includes explicit expressions for the steady-state concentrations applicable to the ERT photochemical mechanism.

At this point, the program passes control to the integration algorithm which is controlled by subroutine DRIVE. DRIVE and its 16 subprograms perform the integration of the finite-differenced, ordinary differential equations where the chemistry and diffusion are coupled for each time step.

The integration method is one developed by Hindmarsh and Byrne (1975) named EPISODE (Experimental Package for the Integration of Systems or Ordinary Differential Equations). The method is a variable-step, variable-order, implicit backwards differencing scheme that is particularly useful for solving stiff systems of equations. The model's ordinary differential equations are quite stiff, and the algorithm is well-suited for this application. The EPISODE package has been modified to take into account the block tri-diagonal form of the model's differential equations, which greatly improves its efficiency.

Program control is returned from DRIVE to subroutine KEMOD2 at every time in the program's update schedule. KEMOD2 checks certain parameters returned by DRIVE to ensure that the integration is proceeding satisfactorily. It prints the concentrations when appropriate. It stores the concentrations for subsequent plotting, and punches the ground concentrations if this option has been selected. Then, it updates the emissions data, sky clearness ratios, photodissociation rates, temperature-dependent reaction rates, and K_z coefficients to the current time. If the current time is less than the stop time, program control returns to DRIVE where the integration is performed for another update interval. This cycle of integration, output, and updating is repeated until the end of the simulation. The final task for the program is the generation of printer-plots of ground concentration histories for the more important species.

5.6 Program Modifications

The KEMOD program is quite complicated; only the most probable modifications will be described here. These modifications relate to changing the chemical mechanism, changing the partitioning of emissions data, increasing the number of emission species, and changing the species' concentrations plotted.

5.6.1 Changing the Chemical Mechanism

The program is currently designed to include up to 40 chemical species, 10 of which must be either specified as steady-state approximations, or treated as constants. Thus, up to 30 species can be integrated. These dimensions are felt to be sufficiently large for any practical application of the model.

The program is presently equipped to handle up to 55 chemical reactions. The number of chemical reactions can easily be changed by changing the dimension, N, in the arrays shown below:

LOCATION	ARRAY NAME
COMMON/CHEM2/	RATKON(N), RATEFF(N)
COMMON/CHEM5/	REACT(20,N)
SUBROUTINE RATES	R(N)
SUBROUTINE ISTATE	R(N)

See Appendix A for the list of subroutines which contain common blocks CHEM2 and CHEM5.

To install a new mechanism or change the present mechanism, the user must supply a set of explicit chemical rate equations in subroutine

RATES. There should be one rate equation for each species that is integrated by the program. The indices of the rate equation must correspond to the indices of the species in the species list. The method of determining the components of the rate equations is shown below for one chemical reaction. The expressions, of course, become quite long when a species is involved in many reactions, but the method is the same.

Reaction: $aC_i + bC_j \xrightarrow{k_m} dC_k + eC_l$

Let C_n = C(N) = concentration of n^{th} species

k_m = R(M) = rate constant of m^{th} reaction

a,b = A,B = reactant stoichiometric coefficients of m^{th} reaction

d,e = D,E = product stoichiometric coefficients of m^{th} reaction

Rate Equations:

```
RATE(I)  =  -R(M)* (C(I)**A) * (C(J)**B)
RATE(J)  =  -R(M)* (C(I)**A) * (C(J)**B)
RATE(K)  =  +R(M) * D * (C(I) ** A) * (C(J) ** B)
RATE(L)  =  +R(M) * E * (C(I) ** A) * (C(J) ** B)
```

NOTE: The reactant stoichiometric coefficients are included only when they are not equal to one.

Once the rate equations are determined for the mechanism, the user must supply the matrix of partial derivatives (i.e., the Jacobian) of the rate equations. The elements of the Jacobian are specified in subroutine JACOB as described below:

A(N,N) = an N by N array with the elements of the Jacobian, where N equals the number of species being integrated.

Let: RATE(I) = f(R,C) = f(R(1),...,R(J),...R(NOREAC), C(1),...,C(K),...,C(NOSPEC))

Where: RATE(I) = rate equation of I^{th} species

R(J) = rate constant of J^{th} reaction

C(K) = concentration of K^{th} species

Then: $$A(I,K) = \frac{\partial f(R,C)}{\partial C(K)}$$

Where: I = 1,...,N

K = 1,...,N

Next, the user must determine if changes need to be made concerning the time-varying reaction rate constants embedded in the program. If the new or revised chemical mechanism has more than two photodissociation rates, the user must explicitly scale the extra photodissociation rates in subroutine RATEHI. For a photodissociation rate of reaction J, which is A times the photodissociation rate of reaction I1, and where I1 is the reaction number of the first independent photodissociation rate input, the user should insert the following two statements in RATEHI:

```
RATKON(J)  =  A RAKON(I1)
RATEFF(J)  =  RATKON(J)
```

The user must, of course, remove the statements in RATEHI which presently determine the rates of reactions numbers 6 and 34 from the photodissociation rates I1 and I2, respectively, if they are not appropriate to the new mechanism.

In addition, if the user wants to include temperature dependent reaction rates, subroutine TEMPR must be modified. The user must replace the current expressions for the temperature dependence of rate constant numbers 21 and 37, with similar expressions for the appropriate reaction numbers. The expression should be formulated on the basis of temperature (T) in degrees Kelvin. If none of the rates are temperature dependent, subroutine TEMPR should be converted to a dummy subroutine.

Lastly, the user should supply a new version of subroutine ISTATE that contains explicit algebraic equations for the steady-state initial concentrations of selected species. To determine the steady-state expression for the I^{th} species, let

$$\text{RATE (I)} = f(R,C) = 0$$

and solve for the I^{th} concentration as a function (g) of the rate constants (R) and the other known concentrations (C').

$$C(I) = g(R,C')$$

One may also choose to read in all of the initial conditions and convert ISTATE into a dummy subroutine.

5.6.2 Changing the Partitioning of Emissions Input Data

As explained in Section 5.1.1, the program presently partitions the nitrogen oxides (NO_x) and aldehyde (ALDE) emissions data to more than one species, respectively. The manner in which this is performed is as follows. The program stores the NO_x emissions as NO (species No. 1) and the aldehyde emissions as HCHO (species No. 6). At each update interval, subroutines UPFLX1 and UPSORC are called to retrieve the emissions data for that time. These subroutines perform the partitioning of the emissions data for KEMOD. Let us use as an example the present partitioning of aldehyde emissions as 40% RCHO (species No. 7) and 60% HCHO. Subroutine UPFLX1 would then include the statements:

```
FLXW1(7)  =  .40*FLXW1(6)
FLXW2(7)  =  .40*FLXW2(6)
FLXW1(6)  =  .60*FLXW1(6)
FLXW2(6)  =  .60*FLXW2(6)
```

Subroutine UPSORC would include the statements:

```
   DO 30  J    =  1,NOSTAT
   PSR1(7,J)  =  .40 PSR1(6,J)
   PSR2(7,J)  =  .40 PSR2(6,J)
   PSR1(6,J)  =  .60 PSR1(6,J)
   PSR2(6,J)  =  .60 PSR2(6,J)
30 CONTINUE
```

If the user wishes to change the partitioning percentages, he/she merely changes the corresponding coefficients shown above.

5.6.3 Increasing the Number of Species With Emissions Inputs

The program's arrays are presently dimensioned for seven emissions species and can easily be modified for up to 17 species. To change the arrays to handle emissions for N species ($7 < N \leq 17$), the following changes should be made:

LOCATION	ARRAY NAME or Statement
COMMON/CHEM5/*	LOCFLX(N)
COMMON/FLUXES/*	FLXIN(N,200)
COMMON/PS1/*	PS(N,5,75),LOCPSF(N)
PROGRAM KEMOD	PTSR(N,200),AF(N)
SUBROUTINE KEMOD2	PTSR(N,200)
SUBROUTINE KEMOD2	CALL XMIT(-N*375,0.0,PS)

Various input/output formats in KEMOD, KEMOD2, UPFLX1, and UPSORC also require modification when the number of emissions species exceeds seven.

5.6.4 Changing the Plot Species

The plotting software is designed to provide a plot of surface concentrations of up to five species. The user may change the number of species plotted (NPTS), the plotting symbols (array KSYM), and the plot species indices (array KPS) by inserting the appropriate statements in subroutine KEMOD2. As an example, suppose the user wishes to plot the 10th and 20th species in the species list with symbols A and B, respectively. The user would insert the statements:

```
DATA KSYM/1HA,1HB/

KPS(1)  =  10

KPS(2)  =  20

NPTS  =  2
```

---

*See appendix A for the list of subroutines with this common block.

and remove the present statements pertaining to these variables. The user will also wish to modify format statement number 1001 in subroutine COPLOT to define the new legend.

6. REFERENCES

ASME 1968. *Recommended Guide for the Prediction of Dispersion of Airborne Effluents*. M. Smith (ed), New York, NY.

Baulch, D.L., D.D. Drysdale, and D.G. Horne 1973. "Rate Constants for Reactions of the H_2-N_2-O_2 System" in Chemical Kinetic Data Survey V. Sixty-six Contributed Rate and Photochemical Data Evaluations on Ninety-four Reactions, NBSIR 73-206. D. Garvin, ed., National Bureau of Standards, Washington, DC, pg 49-115.

Briggs, G.A. 1969. *Plume Rise*. AEC Critical Review Series.

Briggs, G.A. 1975. *Plume Rise Predictions*. Lectures on Air Pollution and Environmental Impact Analysis, American Meteorological Socity, Boston, MA, September.

Demerjian, K.L., K.L. Shere, and J.T. Peterson 1979. "Theoretical Estimates of Actinic (Spherically Integrated) Flux and Photolytic Rate Constants of Atmospheric Species in the Lower Troposphere." *Advances in Environmental Science and Technology*, J.N. Pitts (ed); Vol 9 (in press).

Hindmarsh, A.C. and G.D. Byrne 1975. *Experimental Package for Integration of Systems of Ordinary Differential Equations (EPISODE)*. Lawrence Livermore Laboratory, UCID-20112.

Lloyd, A.C., F.W. Lurman, D.G. Godden, J.F. Hutchins, A.Q. Eschenroeder, R.A. Nordsieck 1979. *Development of the ELSTAR Photochemical Air Quality Simulation Model and Its Evaluation Relative to the LARPP Data Base*, ERT Report 5287 prepared for Coordinating Research Council, New York (in press).

Moortgat, G.EK. F. Slemr, W. Seiler, and P. Warneck 1978. *Chem. Physics LeH.*, in press.

Peterson, J.T. 1977. *Dependence of NO_2 Photodissociation Rate Constant on Altitude*, Atmospheric Environment, February.

Peterson, J.T., K.L. Demerjian, and K.L. Shere 1976. *Actinic Solar Flux and Photolytic Rate in the Troposphere*, International Conference on Photochemical Oxidant Pollution and its Control.

Turner, D.B. 1969. *Workbook of Atmospheric Dispersion Estimates*. U.S. Department of Health, Education & Welfare. Public Health Service Publication No. 999, AP-26.

APPENDIX A

COMMON BLOCK LOCATIONS IN THE CODE

1. Meteorological Module Common Blocks

Common Block	Location
AIRQAL	GETCON, METIN, METMOD, WINDY
CELECT	EDDY, STABIL
CNTRL	CROSIT, METMOD, SETTUP, WINDY
CORIOL	EDDY, STABIL
CROSS	CROSIT
DATES	GETAZV, METIN, METMOD, WINDRD, WINDY,
DIFDAT	EDDY, STADIF, KEKLAY
GRID	CROSIT, EDGE, METMOD, PLACIT, SETIN, SETTUP, WINDY
IMAG	SETPLT
INPUTS	CROSIT, EDDY, GETCON, METIN, METMOD, SETTUP, WINDY
KZINPT	EDDY, KZDATA, METIN, METMOD, WINDY
ORIGIN	BARIER, CROSIT, EDDY, GETAZV, METMOD, RUFNES, SETIN, SETTUP, WINDY
PEVE	EDDY, RUFNES, UXYPOS
REUSE	CROSIT, GETCON, METIN, METMOD, WINDY
TEMPHT	EDDY, KEKLAY, SMOOTH, STADIF
TRAJ	CROSIT, EDDY, METMOD, PLACIT, SETTUP, UXYPOS, WINDY
VGRID	CROSIT, BLOCK DATA
WDATA	ANGTST, AZVDIS, GETAZV, GETCON, METIN METMOD, SETIN, SETTUP, WINDY, WSECLS
WHERE	EDDY, METIN, METMOD, SETIN, SKY, WINDY
WIND	CROSIT, EDDY, METMOD, PLACIT, SETTUP, UXYPOS, WINDY
WINFLD	GETAZV, GETCON, METIN, METMOD, WINDY

2. Emissions Module Common Blocks

Common Block	Locations
ANGLES	LOCATE, PONTEM, SEGMET
ANSWER	AREAEM, EMMOD, GRIDIT, PLUMAS, PONTEM, SEGMET, STACKS
DEGRID	AREAEM, GRIDIT
EMRATE	AREAEM, EMMOD
HISTRY	ADHOUR, AREAEM, EMMOD
INPUT	AREAEM, EMMOD, PONTEM, STACKS
LABLIN	AREAEM, EMMOD, PONTEM, STACKS
LABOUT	AREAEM, EMMOD, PONTEM, STACKS
LINKS	DISTAN, LOCATE, PONTEM, SEGMET
MXHITE	EMMOD, PONTEM
ORIGIN	EMMOD, PONTEM, STACKS
PARCEL	AREAEM, EMMOD, PLUMAS, PONTEM, STACKS
PASSES	PONTEM
PSDAT	PONTEM, STACKS
SIGMAS	PARTIT, PLUMAS, PONTEM, SEGMET, BLOCK DATA
SIGTAL	PARTIT, PLUMAS, BLOCK DATA
TABLES	ADHOUR, AREAEM, EMMOD, PONTEM
TEMPS	EMMOD, PONTEM
TRAJ	EMMOD, LOCATE, PONTEM
WIND	DISTAN, EMMOD, LOCATE, PONTEM, SEGMET
WORKER	ADHOUR, AREAEM, GRIDIT, PONTEM, STACKS

3. Chemical-Diffusion Module Common Blocks

Common Block	Locations
CHEM1	DIFFUN, DRIVE, ISTATE, KEMOD2, PEDERV, PSET, RATES, STEADY, TSTEP
CHEM2	DIFFUN, DRIVE, ISTATE, KEMOD2, PEDERV, PSET, RATEHI, RATES, STEADY, TEMPR, TSTEP
CHEM3	DIFCOF, DIFFUN, DRIVE, KEMOD2, PEDERV, PSET, TSTEP, UPFLX1
CHEM4	KEMOD2, PHOTOD, UPRAT2
CHEM5	DIFFUN, KEMOD2, UPFLX1
EPCOM1	ADJUST, COSET, DRIVE, INTERP, PSET, TSTEP
EPCOM2	DRIVE, PSET, TSTEP
EPCOM3	DRIVE, PSET, TSTEP
EPCOM4	DRIVE, PSET, TSTEP
EPCOM5	DRIVE, PSET, TSTEP
EPCOM6	DRIVE, KEMOD, KEMOD2, PSET, TSTEP
EPCOM7	DRIVE, PSET, TSTEP
EPCOM8	DRIVE, PSET, TSTEP
EPCOM9	DRIVE, PSET, TSTEP
EPCM10	ADJUST, COSET, TSTEP
EPCM12	DRIVE, PSET, TSTEP
FLUXES	KEMOD, KEMOD2, UPFLX1
INPUTS	KEMOD, KEMOD2, PHOTOD
PRPLOT	COPLOT, KEMOD2
PS1	DIFFUN, KEMOD, KEMOD2, UPSORC
PS2	DIFFUN, KEMOD2, UPSORC

APPENDIX B

CODE SUBROUTINE DESCRIPTIONS

1. Meteorological Module Subroutines and Their Use

PROGRAM METMOD

Use. METMOD is the main program for the meteorological module of the code. It reads several key control parameters that determine which submodules are to be called. These parameters specify whether the wind trajectory is to be generated from wind data or is specified in the input data-cards, whether or not air quality data is to be interpolated along the trajectory, and whether or not diffusivity coefficients (K_z) are to be computed. In addition, METMOD controls the multiple case feature of the program.

SUBROUTINE ANGTST (XS,YS,NGOOD,KUSE,OK)

Use. Compares the azimuth from (XS,YS) to the new station against the corresponding azimuths for any previously qualified stations. If any two azimuths are less than 2 degrees apart, ANGTST returns a positive flag; otherwise, it returns a negative flag. ANGTST is called by GETAZV.

Arguments. XS = x-coordinate of current position
YS = y-coordinate of current position
NGOOD = number of measurement stations qualified thus far
KUSE = index of new candidate measurement station
OK = output flag. If positive, angle test is OK (no other station within 2 degrees); if negative, test not OK

SUBROUTINE AZVDIS (NP,AZV,DDD,AZM,VEL,IRF)

Use. Given raw azimuth, velocity, and distance data from up to three measurement stations, AZVDIS combines them, according to the prescribed interpolation rule, to produce a single azimuth and velocity. AZVDIS is called by WINDY.

Arguments. NP = number of measurements to combine (up to three)
AZV = array containing raw azimuths and velocities
DDD = array with distances from current position to closest stations
AZM = calculated distance
VEL = calculated velocity
IRF = interpolation flag; if 1, inverse distance weighting is used; if 2, inverse distance squared weighting is used

SUBROUTINE BARIER (X,Y,XS,YS,OK)

Use. Checks whether or not two given points are on the same side of each of a set of (up to 20) finite-length, straight-line barriers. The barrier line segments are stored internally, being defined by the coordinates of their end points. BARIER is called by GETAZV and WINDY.

Arguments. X = x-coordinate of point of interest
Y = y-coordinate of point of interest
XS = x-coordinate of reference point
YS = y-coordinate of reference point
OK = output flag; if positive, both points are on the same side of all barriers; otherwise, the points are on opposite sides of at least one barrier

SUBROUTINE CELAVG (DH,NMESHP,CELVAT,AKZ,NZHI,STORE)

Use. This routine computes average K_z coefficients for use in the chemical-diffusion module from the detailed internal K_z profile. CELAVG is called by EDDY.

Arguments. DH = vertical increment for internal K_z profile
NMESHP = number of vertical mesh points (up to five)
CELVAT = elevations of vertical mesh points
AKZ = array of internal K_z coefficients
NZHI = number of K_z coefficients in AKZ array
STORE = array of average K_z coefficients returned

SUBROUTINE CROSIT

Use. CROSIT generates trajectory grid square crossing schedules and plots the wind trajectory. CROSIT is called by WINDY.

Arguments. None

SUBROUTINE DATE (IOLD,MOVE,IWAY,NEW)

Use. Calculates a new time and date from an old time and date and a prescribed time increment. Time is on a 0-2400 basis and dates are represented as year, month, day (e.g., 780901 for Sept. 1, 1978). DATE is called by WINDY.

Arguments.
IOLD = 2-word array with time and date to be changed
MOVE = time increment in minutes
IWAY = if positive, time advances; otherwise, time is decreased
NEW = 2-word output array with new time and date

SUBROUTINE DIFFUS (STAB,ZHT,AKZ,USTAR)

Use. This routine determines the K_z coefficient at a specific elevation within the surface layer. DIFFUS is called by STADIF.

Arguments.
STAB = Monin-Obukhov length
ZHT = elevation above ground
AKZ = value of K_z returned
USTAR = friction velocity

FUNCTION DOTTY (A,B)

Use. DOTTY computes the dot product of the vectors A and B. DOTTY is called by ANGTST.

Arguments.
A = a 3-component vector
B = a 3-component vector

FUNCTION DTIME (A)

Use. This function converts time from the 0-2400 clock to hours and decimal hours. DTIME is called by EDDY.

Argument. A = time on a 0-2400 hour clock

SUBROUTINE EDDY (KPWIND)

Use. This routine is the driver for the eddy diffusivity submodule. EDDY determines a schedule of atmospheric stability classes for the emissions module and a schedule of diffusion (K_z) coefficients for the chemical-diffusion module. EDDY is called by WINDY.

Argument. KPWIND = Hollerith variable equal to YES when punched output is desired

SUBROUTINE EDGE (X,Y,ITF,IEDGE)

Use. EDGE checks whether or not the point X,Y is outside the modeling region. EDGE is called by WINDY.

Arguments.
X = x-coordinate of point of interest
Y = y-coordinate of point of interest
ITF = direction flag; if positive, the trajectory is proceeding forward; if negative, the trajectory is proceeding backwards
IEDGE = output flag; if zero, the point is inside the modeling region; if non-zero, the point is outside the modeling region with the sign of IEDGE indicating the direction of motion

SUBROUTINE EXTRP (IHOUR,ARY,VAL)

Use. This routine seeks a value corresponding to IHOUR in the 24-word data array ARY, assuming that the values in ARY correspond to clock times of 0000, 0100, 0200, etc. If positive or zero, the value found is simply returned. If the desired value is missing (indicated by a negative entry), EXTRP will use linear interpolation to fill the gap, provided the two adjacent values are non-negative. Failing that, EXTRP will attempt to fill the gap by extrapolation, provided that the gap is no more than two locations wide and that two consecutive non-negative values exist on one side of the gap. A negative value returned indicates that the data gap could not be filled by either method. EXTRP is called by GETAZV.

Arguments. IHOUR = clock time on a 0-2400 basis
ARY = array of 24 data values one hour apart
VAL = data value selected or calculated to correspond to IHOUR

FUNCTION FULGOL (A,ZO)

Use. This function determines inverse Monin-Obukhov length (1/L). FULGOL is called by STABIL.

Arguments. A = Fulle atmospheric stability parameter
ZO = surface roughness length (meters)

SUBROUTINE GETAZV (KOLD,ID,XS,YS,NC,AZV,NGOOD,DDIS)

Use. This routine attempts to retrieve measurement records from the desired number of neighboring wind measurement stations having valid data for the given date and time. GETAZV is called by WINDY.

Arguments. KOLD = 2-word array containing time (on a 0-2400 basis) and date (year, month, day)
ID = 2-word array containing the station identification (may be blank)
XS = x-coordinate of current position
YS = y-coordinate of current position
NC = desired number of close stations (up to 3)
AZV = array containing an azimuth and velocity for each close station found
NGOOD = number of close stations found
DDIS = array containing distances from current position to close stations

SUBROUTINE GETCON (KNEW,NC,IRF)

Use. This routine interpolates and extrapolates measurement station data stored in the array CON to a trajectory node location. Surface temperature and air quality data are normally stored in the array CON. GETCON is called by WINDY.

Arguments. KNEW = 2-word array with the time and date
NC = desired number of stations (up to 3)
IRF = variable equaling 1 or 2 indicating the exponent in the interpolation formula, 1/Distance** IRF

SUBROUTINE JULIAN (IDATE,JULDAT)

Use. This routine converts a 6-character date (YR-MO-DA) to a Julian date. JULIAN is called by METIN.

Arguments. IDATE = 6-character date
JULDAT = a 2-word array of Julian day and year

SUBROUTINE KEKLAY (LMAX,ZMAX,ZHIGH,AKMIN,DH)

Use. This routine determines the K_z coefficients in the Ekman layer. KEKLAY is called by EDDY.

Arguments. LMAX = number of points in the verical temperature sounding
ZMAX = height of maximum potential temperature
ZHIGH = height of highest point in temperature sounding
AKMIN = minimum K_z coefficient allowed
DH = vertical increment for the internal K_z profile

FUNCTION KLASS (A)

Use. This function returns the atmospheric stability class. KLASS is called by STADIF.

Arguments. A = Fulle atmospheric stability parameter

SUBROUTINE KZDATA (IERR)

Use. This routine reads the input data for the eddy diffusivity submodule. KZDATA is called by EDDY.

Argument. IERR = output flag which is returned with a positive value when an error is encountered in reading the data

SUBROUTINE METIN (IDT,LTAPE)

Use. This subroutine retrieves wind, surface temperature, air quality, and solar radiation data from the RAMS data tape for one day. METIN edits, stores and computes regional averages of various atmospheric parameters. METIN is called by WINDY.

Arguments. IDT = 6-character date (YR-MO-DA)
LTAPE = logical unit number of the peripheral device with the meteorological data

SUBROUTINE PLACIT (DT,L,VX,VY,POS,NDX)

Use. PLACIT computes the coordinates and grid square indices of a new trajectory point. PLACIT is called by CROSIT.

Arguments. DT = time duration of this trajectory segment, minutes
L = index of last trajectory point in trajectory data arrays
VX = x-component of V(L) (input)
VY = y-component of V(L) (input)
POS = 2-word array containing x and y coordinates of new position
NDX = 2-word array containing grid square indices (I,J) of new point

SUBROUTINE RUFNES (ZO,J)

Use. This routine determines the surface roughness length at a particular location. RUFNES is called by EDDY.

Arguments. ZO = surface roughness length (meters)

J = integer corresponding to the hour of the day plus one

SUBROUTINE SETIN

Use. This routine initializes the names and coordinates of all measurement stations. SETIN also designates the total number of measurement stations. SETIN is called by WINDY.

Argument. None

SUBROUTINE SETTUP

Use. SETTUP serves as the initialization routine for the generation of trajectory grid square crossing schedules and the trajectory printer-plot. SETTUP is called by CROSIT.

Argument. None

SUBROUTINE SKY (UV,CLOUDY,IDATE)

Use. SKY generates sky clearness ratios from ultraviolet radiation data. SKY is called by METIN.

Arguments. UV = a 24-word array of hourly ultraviolet radiation data in Langley's per minute

CLOUDY = a 24-word array of hourly sky clearness ratios used in the chemical-diffusion module

IDATE = 6-character date (YR-MO-DA)

SUBROUTINE SMOOTH (ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)

Use. This routine smooths the raw vertical temperature profile and divides the profile into layers on the basis of lapse rates. SMOOTH is called by EDDY.

Arguments. ZW = height at which surface wind data was measured (meters)

ZO = surface roughness length (meters)

USFC = wind velocity measured at Z = ZW

ZHIGH = height of highest point in vertical temperature profile

DH = vertical increment for internal K_z profile

AKMIN = minimum value of K_z (m^2/min)

ISKLAS = atmospheric stability class on a scale of 1 to 6 corresponding to Pasquill-Gifford classes A to F

SUBROUTINE STABIL (STAB,J,A,ZTOP,DUDZ,DELT,U,ZHTMIX,USTAR,ZW,ZO)

Use. This routine determines the Fulle stability parameter, the Monin-Obukhov length, and the internal mixing layer height from the vertical temperature gradient within an atmospheric layer. STABIL is called by STADIF.

Arguments.
STAB = Monin-Obukhov length (L)
J = index for the atmospheric layer
A = Fulle atmospheric stability parameter
ZTOP = highest elevation in the atmospheric layer
DUDZ = vertical wind shear
DELT = vertical temperature gradient in the atmospheric layer
U = surface wind velocity
ZHTMIX = calculated mixing layer height
USTAR = friction velocity
ZW = height at which surface wind data was measured (meters)
ZO = surface roughness length (meters)

SUBROUTINE STADIF (ZW,ZO,USFC,ZHIGH,DH,AKMIN,ISKLAS)

Use. This routine controls the calculation of stability and diffusivities. STADIF is called by EDDY.

Arguments.
ZW = height at which surface wind data was measured (meters)
ZO = surface roughness length (meters)
USFC = surface wind velocity

ZHIGH = height of highest point in temperature sounding

DH = vertical increment for the internal K_z profile

AKMIN = minimum value of K_z

ISKLAS = atmospheric stability class on a scale of 1 to 6 corresponding to Pasquill-Gifford classes A to F

FUNCTION TIMIN (T)

Use. Converts time from 0-2400 basis to minutes past midnight (0-1440). TIMIN is called by CROSIT, SETTUP, and WINDY.

Argument. T = clock time on a 0-2400 basis

FUNCTION UNIDOT (A,B,)

Use. Returns the dot product of a unit vector parallel to A with a unit vector parallel to B, i.e., the cosine of the acute angle separating the vectors A and B. UNIDOT is called by BARIER.

Arguments. A = a 3-component vector
B = a 3-component vector

SUBROUTINE UNITV (A,B)

Use. Returns B, the unit vector parallel to vector A. UNITV is called by ANGTST.

Arguments. A = a 3-component vector
B = a 3-component unit vector parallel to A

SUBROUTINE UXYPOS

Use. This routine establishes the wind speed and location information used by EDDY. UXYPOS is called by EDDY.

Argument. None

SUBROUTINE WINDRD (KPWDAT,LTAPE)

Use. This subroutine is used in non-St. Louis applications to read the wind measurement data. For the St. Louis application, a RETURN statement has been inserted near the beginning of the routine so that it acts like a dummy subroutine. WINDRD is called by WINDY.

Arguments. KPWDAT = a Hollerith variable equaling YES when it is to print the wind data read from cards

LTAPE = logical unit of the peripheral device with the wind data. Use LTAPE = 3 for cards.

SUBROUTINE WINDY (IWAS,LTAPE)

Use. WINDY is the secondary driver of the meteorological module. This routine generates the air trajectories (backward or forward) from a prescribed start location and time. WINDY is called by program METMOD.

Arguments. IWAS = flag which is equal to zero on the first call and greater than zero on subsequent calls

LTAPE = logical unit number of the peripheral device with the meteorological data

SUBROUTINE WSECLS (X,Y,DIS,IDIS)

Use. This routine calculates squared distances from the point X,Y to each measurement station in SSTAN, and then orders the stations by increasing distance out to a maximum value (RMAX). WSECLS is called by GETAZV and WINDY.

Arguments. X = x-coordinate of current position

Y = y-coordinate of current position

DIS = array of squared distances from X,Y to all measurement stations. Those values of DIS less than RMAX are arranged in increasing order at the top of the table

IDIS = array of station indices corresponding to entries in the DIS array. (The indices identify the stations in the SSTAN array).

BLOCK DATA

Use. This block data program contains the variable size grid square identification numbers distributed to a fixed size grid square array, IVG. The current grid square identifiers correspond to grid squares varying in size from 1 x 1 kilometer to 10 x 10 kilometers in a St. Louis grid with origin at UTM coordinates x = 680 (km) and y = 4230 (km). The array (IVG) has been dimensioned 100 by 100 corresponding to a 100 kilometer square region considered as the emissions grid. In the portions of the region where grid square identifiers are absent, the array contains zeroes.

2. Emissions Module Subroutines and Their Use

PROGRAM EMMOD

Use. EMMOD is the main program for the emissions module. It reads all the card-image data into the proper arrays and controls the over-all generation of source emissions schedules along trajectories.

Note. When modified with computer installation specific control statements for rewinding and/or skipping files, the program can be used for multiple trajectories on the same day or different days.

SUBROUTINE ADHOUR

Use. This routine inserts even hour values in an arbitrary temporal schedule and interpolates variables associated with the schedule to the even hours. ADHOUR is called by EMMOD.

Argument. None

SUBROUTINE AREAEM (LUAREA,IPUNCH,KOK)

Use. This routine reads the RAPS area source emissions file and determines the area source emissions schedule along the trajectory. AREAEM is called by EMMOD.

Arguments. LUAREA = logical unit of peripheral device with area source emissions data

IPUNCH = a Hollerith variable equal to YES when punched output is desired

KOK = a completion flag which is returned with a positive value if an error is encountered

SUBROUTINE DHPLUM (TA,TS,U,ISTAB,ACMM,HTINV,HTPS,DH,PENTFR,DTDZ)

Use. This subroutine calculates the plume rise and the inversion layer penetration fraction for a point source. DHPLUM is called by PONTEM.

Arguments.
TA = ambient temperature (°C)
TS = stack effluent temperature (°C)
U = wind speed (meters/second)
ISTAB = atmospheric stability class
ACMM = stack effluent volumetric flow rate (cubic meters/minute)
HTINV = inversion base height above surface (meters)
HTPS = stack height (meters)
DH = plume rise (meters)
PENTFR = inversion penetration fraction (0. to 1.)
DTDZ = potential temperature gradient of stable layer (°C/meter)

FUNCTION DISTAN (IPT,TS,TE)

Use. This function returns the distance along a trajectory between two times. DISTAN is called by PLUMAS.

Arguments.
IPT = index of the last trajectory node
TS = initial time (minutes)
TE = final time (minutes)

FUNCTION ERF (X)

Use. ERF returns the value of the error function of X. ERF is called by PLUMAS and PARTIT.

Argument. X = the independent variable

SUBROUTINE GESTAB (TIME,ISTAB,DTNEXT,NXSTAB)

Use. This routine determines the atmospheric stability at the specified time. In addition, the value of and time difference to the next stability class change are determined. GESTAB is called by PONTEM and PLUMAS.

Arguments.
TIME = current time (minutes)
ISTAB = current stability class
DTNEXT = time difference to next stability class change
NXSTAB = next stability class value

SUBROUTINE GRIDIT (LUGRID,KOK)

Use. This subroutine reads the RAPS area source grid identifiers and corresponding areas. GRIDIT is called by EMMOD.

Arguments.
LUGRID = logical unit peripheral device with RAPS grid description
KOK = a completion flag which is returned positive if an error is encountered

SUBROUTINE LOCATE (IPT,XP,YP,PDIST,TPASS,KFLAG,ISTAB)

Use. This routine examines the location of a point source relative to a trajectory segment. It determines whether or not the point source is passed on a particular trajectory segment and whether or not it is close enough to be considered for the point source emissions schedule. LOCATE is called by PONTEM.

Arguments.
IPT = index of the last trajectory node
XP = point source x-coordinate (km)
YP = point source y-coordinate (km)
PDIST = perpendicular distance to XP, YP
TPASS = time source is passed (minutes)
KFLAG = a variable which is positive when a point source's location meets all criteria to be included on the segment
ISTAB = atmospheric stability class

SUBROUTINE PARTIT (X,HT,HTINVR,HTINTF,NOSTAT,ISTAB,TALL,PENTFR,VFRACT)

Use. This routine is used to distribute the elevated point source's emissions to vertically partitioned atmospheric layers. PARTIT is called by PONTEM.

Arguments.
X = downwind distance from source (km)
HT = height of source above surface (meters)
HTINVR = height of inversion base or mixing depth (meters)
HTINTF = vector of heights of layer interfaces (meters)
NOSTAT = number of layers
ISTAB = atmospheric stability class
TALL = Hollerith variable indicating whether or not the source is a "tall stack" (YES or NO)
PENTFR = inversion penetration fraction (0. to 1.)
VFRACT = a vector of the normalized vertical distribution of the source's entrained emissions

SUBROUTINE PLUMAS (IPT,TPASS,PDIST,PLUMFR,ISTAB,DXFREZ,TALL,KPLAG)

Use. PLUMAS is used to determine the fraction of point source's plume which is present in the Lagrangian parcel after a fixed time interval from interception of the plume. PLUMAS is called by PONTEM.

Arguments.
IPT = index of the current trajectory node
TPASS = input as time of day when source is passed, and returned as time of day when plume is intercepted (minutes)
PDIST = perpendicular distance to source at TPASS (km)
PLUMFR = fraction of plume's mass present in the parcel after the fixed time interval
ISTAB = atmospheric stability class

DXFREZ = a fixed time interval between the plume interception and the lateral distribution evaluation

TALL = a Hollerith variable indicating whether or not the source is a "tall stack" (YES or NO)

KPLAG = a flag returned with a value corresponding to the characteristics listed below:

- -1 = plume's lateral spread is too slow to reach the parcel (source ignored)
- 1 = plume's lateral diffusion out of the parcel is dominant
- 2 = plume's lateral diffusion into the parcel is dominant

SUBROUTINE PONTEM (LUPONT,IPUNCH,KOK,NDSUMS)

Use. This routine controls the generation of point source emission schedules along a trajectory. PONTEM is called by EMMOD.

Arguments. LUPONT = logical unit of peripheral device with point source emissions data

IPUNCH = a Hollerith variable equal to YES when punched output is desired

KOK = a completion flag which is returned with a negative value when errors are encountered

NDSUMS = a Hollerith variable equal to YES when regional sums of the point source emissions inventory are desired

SUBROUTINE SEGMET (P)

Use. This routine computes the length and bisector angles for each segment of the trajectory. In addition, it computes the maximum radii and perpendicular distance criteria used in evaluating which point sources will have plumes potentially intersecting the trajectory. SEGMET is called by PONTEM.

Argument. P = vector of trajectory node locations

SUBROUTINE STACKS (TIME,LUPONT,KOK,IWAS,NDSUMS)

Use. This routine reads the RAPS point source emissions file each hour and determines the regional point source emission totals if desired. STACKS is called by PONTEM.

Argument. TIME = time of day (minutes)
LUPONT = logical unit of peripheral device with the point source emissions data
KOK = a flag which returns with a negative value if errors are encountered
IWAS = a flag which is input as zero on the first call to the subroutine and input as one on all subsequent calls
NDSUMS = a Hollerith variable equal to YES when regional sums of the point source emissions inventory are desired

FUNCTION TRPLOG (X,XS,FS,NXF,NF)

Use. This function performs log-log interpolation of tabular functions. TRPLOG is called by PLUMAS and PARTIT.

Arguments. X = value of independent variable
XS = vector of independent variable values corresponding to the tabulated function values
FS = array of tabulated function values
NXF = number of tabulated function values
NF = number of functions in FS array (normally 1)

BLOCK DATA

Use. The emissions module block data program contains Gaussian dispersion coefficients used in the point source submodule. The σ_y and σ_z coefficients for typical stack heights are tabulated by atmospheric stability class and distance. For tall stacks, the coefficients a, b, c, and d used in the equations

$$\sigma_y = ax^b$$

$$\sigma_z = cx^d$$

are included as functions of atmospheric stability class.

3. Chemical-Diffusion Module Subroutines and Their Use

PROGRAM KEMOD

Use. KEMOD is the main program for the chemical-diffusion module. This program reads the emissions data and calls KEMOD2 to perform the remainder of the input/output and program control.

SUBROUTINE ADJUST (Y,NO)

Use. This routine is a part of the EPISODE integration package. It is used to adjust the concentrations history array upon a reduction of the order of the approximation. ADJUST is called by TSTEP.

Arguments.
Y = a 2-dimensional array containing the concentrations history
NO = the maximum number of differential equations

SUBROUTINE BLKDEC (QB,NN,KK,IP,A,C,TL,W,NOGO,BCFLAG)

Use. This routine performs LU decomposition of a block-tridiagonal matrix system of linear equations. BLKDEC is called by PSET.

Arguments.
QB = a 3-dimensional array containing the diagonal submatrices
NN = order of diagonal submatrices
KK = block order of the system
IP = a 2-dimensional array of order NN*KK used for storing pivot information
A = a vector of (KK-1) coefficients representing the lower diagonal submatrices which are scaled identify matrices in this problem
C = a vector of (KK-1) coefficients representing the upper diagonal submatrices which are scaled identity matrices in this problem

TL = a 3-dimensional array used to store the lower diagonal elements of Q

W = a work vector of length NN

NOGO = a completion code which is returned as one when a singular matrix is encountered

BCFLAG = an integer vector of surface boundary condition codes

Remarks. The matrix is of the form

$$Q = \begin{bmatrix} \mathrm{QB}(1) & \mathrm{C}(1) & o & o & o \\ \mathrm{A}(1) & \mathrm{QB}(2) & \mathrm{C}(2) & o & o \\ o & . & . & . & o \\ o & o & . & . & \mathrm{C}(\mathrm{KK}-1) \\ o & o & o & \mathrm{A}(\mathrm{KK}-1) & \mathrm{QB}(\mathrm{KK}) \end{bmatrix}$$

BLKDEC is designed only to be used with BLKSOL.

SUBROUTINE BLKSOL (QB,NN,KK,RHS,IP,A,C,TL,W,BCFLAG)

Use. This routine solves a block-tridiagonal matrix system of linear equations QX = RHS after LU decomposition has been performed on Q. BLKSOL is called by TSTEP.

Arguments. Same as BLKDEC except RHS and deletion of NOGO.

RHS = input as the right-hand side vector and returned as the solution vector

SUBROUTINE CHECKY (T,Y,YO,N,KOK)

Use. This routine is used to check for negative concentrations. CHECKY is called by DRIVE.

Arguments.

T = time of day (minutes)

Y = a two-dimensional array containing the concentrations history

YO = the vector of concentrations last passed to subroutine DRIVE from KEMOD2

N = the number of differential equations

KOK = a completion code which is returned with a negative value when negative concentrations are encountered

SUBROUTINE COPLOT (NDPT)

Use. This routine controls the generation of a printer-plot of up to five ground concentrations versus time.

Argument. NDPT = the number time increments to plot

SUBROUTINE COSET

Use. This routine is a part of the EPISODE package. It is used to set coefficients for the predictor-corrector formula. COSET is called by TSTEP.

Argument. None.

SUBROUTINE DEC (N,NDIM,A,IP,IER)

Use. This subroutine performs the LU decomposition of a matrix. DEC is called by BLKDEC.

Arguments.

N = order of the matrix A

NDIM = declared dimension of the matrix A

A = on input, a matrix to be triangularized; on output, A contains both the upper and lower elements of the L and U matrices

IP = a vector of pivot indices

IER = a flag returned nonzero when A is a singular matrix

Remarks. DEC is designed only to be used with SOL.

SUBROUTINE DIFCOF (NOSTAT)

Use. This routine calculates the scaled eddy diffusivity coefficients used in the finite difference form of the diffusion equation. DIFCOF is called by KEMOD2.

Argument. NOSTAT = number of vertical mesh points

SUBROUTINE DIFFUN (N,T,Y,YDOT)

Use. DIFFUN defines the finite difference ordinary differential equations (YDOT = F[Y,T]). It computes the boundary condition fluxes at time T and the YDOT vector. DIFFUN is called by TSTEP.

Arguments.
N = number of differential equations
T = time of day in minutes
Y = a 2-dimensional array containing the concentrations history
YDOT = the rate of change of Y with respect to time

SUBROUTINE DRIVE (N,TO,HO,YO,TOUT,EPS,IERROR,MF,INDEX,BIGSTP,KOK)

Use. DRIVE is the driver subroutine for the EPISODE package. It performs initialization and control functions for the EPISODE package.

Arguments.
N = number of differential equations
TO = initial time (minutes)
HO = initial time step size (minutes)
YO = initial concentration vector
TOUT = time at which output is desired
EPS = integration error control criteria
IERROR = error control method
MF = method flag which must equal 21 for this version of EPISODE
INDEX = EPISODE input control parameter and output completion code
BIGSTP = maximum time step size allowed
KOK = a completion code returned as negative if an error is encountered

SUBROUTINE FMAX (A,N,B)

Use. This routine determines the maximum value of the elements of the vector A. FMAX is called by COPLOT.

Arguments. A = vector
N = number of elements of A ($N \geq 2$)
B = maximum value

SUBROUTINE INTERP (TOUT,Y,NO,YO)

Use. INTERP is used by the EPISODE package to interpolate the values of Y to the desired output time. INTERP is called by DRIVE.

Arguments. TOUT = desired output time (minutes)
Y = a 2-dimensional array containing the concentrations history
NO = the number of differential equations
YO = a vector of the concentrations at T = TOUT

SUBROUTINE ISTATE

Use. This routine calculates initial conditions for 17 species based on the steady-state assumption. ISTATE is called by KEMOD2.

Arguments. None

Remarks. This routine is specific to the ERT 4.1.78 photochemical mechanism.

SUBROUTINE KEMOD2 (INTIM,TSTOP,NUMFLX,FLXUNT)

Use. KEMOD2 is the secondary driver for the chemical-diffusion module. It controls all the input/output except for the emissions data. It calls the EPISODE integration package and controls all the updating processes. KEMOD2 is called by program KEMOD.

Arguments. INTIM = initial time in minutes
TSTOP = final time in minutes
NUMFLX = number of species with area source emissions input
FLXUNT = a Hollerith variable equal to MOLE if the emissions data is input in mole fraction instead of mass fraction

SUBROUTINE MATMUL (A,B,C,N,M,L)

Use. This routine computes the matrix product C = A x B.

Arguments.
A = matrix of order N x M
B = matrix of order M x L
C = A x B
N = number of rows of A
M = number of columns of A and rows of B
L = number of columns of B

FUNCTION ITHOUR (TMIN)

Use. ITHOUR converts time of day in minutes to time on the 2400-military clock.

Arguments. TMIN = time of day in minutes

SUBROUTINE JACOB (A,C,R,N)

Use. JACOB computes the Jacobian matrix of the chemical rate equations having the form $dc_i/dt = f(c_1, c_2, \ldots, c_n)$, where c_i is the concentration of the i^{th} species. Each element of the Jacobian is defined by $A_{ij} = \partial f_i/\partial c_j$, i, j = 1, 2, ..., N.

Arguments.
A = Jacobian matrix of size N x N
C = vector of species concentrations
R = vector of chemical rate constants
N = order of the Jacobian matrix

Remarks. This routine is specific to the ERT photochemical mechanism of 4.1.78.

SUBROUTINE PEDERV (NO,Y,Q,NKM)

Use. This routine computes the block-tridiagonal Jacobian matrices by calling JACOB for each vertical cell. PEDERV is called by PSET.

Arguments. NO = number of differential equations

Y = a 2-dimensional array containing the concentrations history

Q = a 3-dimensional array containing the block-diagonal Jacobian submatrices

NKM = number of species being integrated

SUBROUTINE PHOTOD (JSTART,JSTOP,NOSTAT,ZEE)

Use. This routine calculates diurnal schedules of elevation dependent photodissociation rates for the following reactions:

$$NO_2 + h\nu \xrightarrow{k_1} O(^3P) + NO$$

$$HCHO + h\nu \left[\begin{array}{l} \xrightarrow{k'_{38}} HCO + H \\ \\ \xrightarrow{k''_{38}} CO + H_2 \end{array} \right.$$

$$k_{38} = k'_{38} + k''_{38}$$

PHOTOD is called by KEMOD2.

Arguments. JSTART = start time on the 2400-clock

JSTOP = stop time on the 2400-clock

NOSTAT = number of vertical mesh points

ZEE = vector of vertical mesh point elevations

SUBROUTINE PSET (Y,NO,CON,MITER,IER)

Use. PSET is a part of the EPISODE package. It computes the coefficients of the matrix P = I - [H/EL(2)]*J where I is an identity matrix, H is the time step size, EL(2) is a corrector coefficient, and J is the block-diagonal Jacobian matrices. PSET is called by TSTEP.

Arguments. Y = a 2-dimensional array containing the concentrations history

NO = number of differential equations

CON = scaling coefficient equal to H/EL(2)
MITER = a flag for the method of determining Jacobian which must equal 1 for this version of PSET
IER = a flag denoting that a singular matrix has been encountered

SUBROUTINE RATEHI (K)

Use. This routine retrieves photodissociation rates for the Kth mesh point and computes other photodissociation rates which are proportional to those rates computed by PHOTOD. RATEHI is called by KEMOD2, RATES, PEDERV, and ISTATE.

Argument. K = index of the vertical mesh point for which the rates are desired

SUBROUTINE RATES (Y,YDOT)

Use. This subroutine computes the chemical rates using explicit rate equations.

Arguments. Y = a 2-dimensional array containing the concentrations history
YDOT = a vector of the rate of change of concentrations with respect to time

Remarks. This routine is specific to the 4.1.78 ERT chemical mechanism.

SUBROUTINE SCALE (Y1,Y2,YB,YT)

Use. This routine computes the minimum and maximum printer-plot boundaries such that the increments have a minimum number of significant figures.

Arguments. Y1 = minimum value to be plotted
Y2 = maximum value to be plotted
YB = minimum value of the plot boundary
YT = maximum value of the plot boundary

SUBROUTINE SKEDUL (X,T,NX,NINT,UPDELT,INTIM,TSTOP,DELTIN,FAIL,INT)

Use. This routine can perform three different transformations of tabulated functions.

1) It transforms a tabulated function with arbitrarily spaced independent variable increments to a tabulated function with evenly spaced independent variable increments.

2) It transforms a tabulated function input on one evenly spaced independent variable increment to a tabular function with a different increment.

3) It performs linear interpolation of a sparsely tabulated function to create a tabulated function on an evenly spaced independent variable increment. SKEDUL is called by KEMOD2.

Arguments.
X = a tabulated function vector
T = an independent variable vector corresponding to X
NX = number of elements of X upon input
NINT = number of elements of X returned
UPDELT = even increment of the independent variable
INTIM = initial value of independent variable
TSTOP = final value of independent variable
DELTIN = even increment of independent variable on input for transformation 2
FAIL = a Hollerith variable equal to YES if an error is encountered in the transformation process
INT = a flag which is input positive when transformation 3 is desired

SUBROUTINE SOL (N,NDIM,A,B,IP)

Use. This routine solves a linear system of equations AX = B where A matrix has been decomposed into lower and upper triangular form by DEC. SOL is called by BLKSOL.

Arguments. N = order of the A matrix
NDIM = dimensioned size of the A matrix
A = the matrix decomposed into lower and upper triangular form by subroutine DEC
B = the right-hand side vector
IP = the pivot vector obtained from DEC

SUBROUTINE SOL2B2 (A,B)

Use. This routine solves a system of two linear equations of the form AX = B.

Arguments. A = a two-by-two matrix of coefficients of the equations
B = input as the right-hand side vector and returned as the solution vector

SUBROUTINE STEADY (Y,N)

Use. A dummy subroutine as used in this version of the model. It can be modified to compute the concentration of species carried in pseudo-stationary state.

Arguments. Y = a 2-dimensional array containing the concentrations history
N = number of differential equations

SUBROUTINE TEMPR (IT,TIME,T)

Use. This routine computes the temperature dependent rate constants associated with the following two chemical decomposition steps:

$$\mathrm{HNO4} \xrightarrow{k_{21}} \mathrm{NO2 + HO2}$$

$$\mathrm{PAN} \xrightarrow{k_{37}} \mathrm{NO2 + RCO3}$$

Arguments. IT = index of the temperature vector associated with the current temperature
TIME = time in minutes
T = a vector of temperatures in degrees Kelvin

SUBROUTINE TIMEX

Use. This is a dummy subroutine as used in this version of the model.

Argument. None

SUBROUTINE TSTEP (Y,NO)

Use. This routine performs one step of the integration of the differential equations. It is the core integrator of the EPISODE package.

Arguments. Y = a 2-dimensional array containing the concentrations history
NO = number of differential equations

SUBROUTINE UNMIXR

Use. This is a dummy subroutine as used in this version of the model.

Argument. None

SUBROUTINE UPFLX1 (TIME,J)

Use. This routine updates the area source emission fluxes.

Arguments. TIME = time in minutes
J = index of the area source array (FLXIN) corresponding to TIME

SUBROUTINE UPRAT2 (T,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)

Use. This routine updates the photodissociation rates to the current time from the schedule of rates created by subroutine PHOTOD. It adjusts the rates by the sky clearness ratio.

Arguments. T = time in minutes
IK1 = index of the RATK1 and RATK2 arrays corresponding to time T
RATEV = a 2-dimensional array in which the current NO_2 and HCHO photolysis rates at the various elevations are stored
NOSTAT = number of vertical mesh points
NVRATE = number of variable rates
CLOUDY = sky clearness ratio (0 to 1)

SUBROUTINE UPSORC (TIME,IPS,NOSTAT,SPEC)

Use. This routine updates the point source emission rates.

Arguments. TIME = time in minutes
IPS = index of the point source emissions array (PS) corresponding to TIME
NOSTAT = number of vertical mesh points
SPEC = a vector of Hollerith species names

4. Utility Library Subroutines and Their Use

SUBROUTINE FMINF (F,NF,FMIN,NMIN)

Use. FMINF finds the minimum value in a linear array and returns both the minimum value and its array index relative to the first value in the array. Called by WSECLS and PONTEM.

Arguments. F = the first location of a sequence to be checked
NF = the number of consecutive locations to be considered (starting with F)
FMIN = the minimum value among the NF consecutive members of the F array
NMIN = the index of FMIN relative to F, counting F as 1

SUBROUTINE MCHAR (IFC,FROM,ITC,TO,NCHR)

Use. MCHAR moves a string of individual characters from one storage location to another. MCHAR is called by SETPLT.

Arguments.
IFC = position of character to move from. The left most character is position 1
FROM = source word
ITC = position of character to move to
TO = destination word
NCHR = number of characters to move

SUBROUTINE MDATE (IDATE)

Use. A dummy subroutine that is located in the code in a position where it would access the machine date.

Argument. IDATE = machine date

SUBROUTINE NEWPAG (TITLE,NSKIP,IDATE)

Use. This routine prints the TITLE as a header on each page of output.

Arguments.
TITLE = an 80-character array containing any user comment or title
NSKIP = number of pages to skip (normally 0)
IDATE = machine date

SUBROUTINE PREDAT

Use. This routine reads the input data cards for each module of the code from logical unit number five. It writes the data card images on the output file, logical unit number six. It also writes the data card images on logical unit number three, and rewinds this file, so that they can be read by the program.

Argument. None

SUBROUTINE SECOND (A)

Use. SECOND is a dummy subroutine which is positioned in the code to access the computer time used from the beginning of the run.

Argument. A = computer time used in seconds

SUBROUTINE SETPLT (A,B,C,D)

Use. SETPLT is the initialization entry for a general purpose printer plot routine. SETPLT is called by SETTUP and COPLOT.

Arguments. A = minimum value on the horizontal (x) axis
B = maximum value on the x-axis
C = minimum value on the vertical (y) axis
D = maximum value on the y-axis

ENTRY PLTPNT (A,B,C)

Use. PLTPNT enters a new data point in the printer-plot array.

Arguments. A = x-coordinate of new data point
B = y-coordinate of new data point
C = a Hollerith character to be plotted at x,y

ENTRY PLTOUT (IFILE)

Use. PLTOUT is the final call in a plotting sequence. It writes the plot to the output file.

Argument. IFILE = the logical unit of output file

SUBROUTINE SOLAR (SLA,SLO,TZ,IY,IM,ID,D,TIME,NV)

Use. SOLAR is used to compute the solar elevation in degrees. SOLAR is called by SKI and PHOTOD.

Arguments. SLA = latitude in degrees (south = minus)
SLO = longitude in degrees (east = minus)
TZ = time zone

IY = year
IM = month
ID = day
D = solar elevation angle in degrees
TIME = time on the 0-2400 hour clock
NV = 5 for solar elevation calculation

SUBROUTINE XMIT (N,A,B)

Use. When N > 0, XMIT transfers N words from A to B. When N < 0, the scalar value of A is transmitted to N words of B. XMIT is called in all three modules by many different subroutines.

Arguments. N = number of words to transfer
A = value of vector (or scalar) to transfer
B = vector receiving new values

APPENDIX C

CODE FORTRAN SOURCE LISTINGS

1. Meteorological Module

	Page
METMOD (Main)	C-3
ANGTST	C-5
AZVDIS	C-6
BARIER	C-7
CELAVG	C-9
CLOCKT	C-9
CROSIT	C-10
DATE	C-14
DIFFUS	C-15
DOTTY	C-16
DTIME	C-16
EDDY	C-16
EDGE	C-23
EXTRP	C-23
FULGOL	C-26
GETAZV	C-26
GETCON	C-31
JULIAN	C-33
KEKLAY	C-34
KLASS	C-35
KZDATA	C-35
METIN	C-37
PLACIT	C-44
RUFNES	C-44
SETIN	C-45
SETTUP	C-47
SKY	C-49
SMOOTH	C-49
STABIL	C-51
STADIF	C-52
TIMIN	C-52
UNIDOT	C-53
UNITV	C-53
UXYPOS	C-53
WINDRD	C-54
WINDY	C-56
WSECLS	C-64
BLOCK DATA	C-65

2. Emissions Module

EMMOD (Main)	C-78
ADHOUR	C-83
AREAEM	C-83
DHPLUM	C-89
DISTAN	C-90
ERF	C-91
GESTAB	C-91
GRIDIT	C-92
LOCATE	C-93
PARTIT	C-94
PLUMAS	C-97
PONTEM	C-98
SEGMET	C-103
STACKS	C-104
TRPLOG	C-107
BLOCK DATA	C-108

3. Chemical-Diffusion Module

	Page
KEMOD (Main)	C-109
ADJUST	C-110
BLKDEC	C-112
BLKSOL	C-114
CHECKY	C-115
COPLOT	C-116
COSET	C-117
DEC	C-119
DIFCOF	C-120
DIFFUN	C-121
DRIVE	C-123
FMAX	C-130
INTERP	C-130
ISTATE	C-131
ITHOUR	C-133
JACOB	C-133
KEMOD2	C-137
MATMUL	C-152
PEDERV	C-152
PHOTOD	C-153
PRODUK	C-156
PSET	C-156
RATEHI	C-158
RATES	C-158
SCALE	C-161
SKEDUL	C-161
SOL	C-163
SOL2B2	C-164
STEADY	C-164
TEMPR	C-165
TIMEX	C-165
TSTEP	C-165
UNMIXR	C-173
UPFLX1	C-173
UPRAT2	C-174
UPSORC	C-175

4. Utility Library

	Page
FMINF	C-176
MCHAR	C-176
MDATE	C-177
NEWPAG	C-177
PREDAT	C-177
SECOND	C-178
SETPLT	C-178
SOLAR	C-182
XMIT	C-183

1. Meteorological Module Listing

```
      PROGRAM METMOD
C
C        THIS IS THE DRIVER PROGRAM FOR THE ERT METEORELOGICAL MODULE.
C
C         SUBROUTINES AND FUNCTIONS  REQUIRED
C
C             ANGTST    AZVOIS    BARIER    CELAVG    CLOCKT    DATE
C             FMINF     FULGOL    GETAZV    GETCON    JULIAN    KEKLAY
C             KZDATA    MCHAR     MDATE     METIN     NEWPAG    PLACIT
C             PREDAT    RUFNES    SECOND    SETIN     SETPLT    SETTUP
C             SKY       SMOOTH    SOLAR     STABIL    STADIF    TIMIN
C             UNIDOT    UNITV     UXYPOS    WINDY     WINDRO    WSECLS
C             XMIT      CROSIT    KLASS     DTIME
C
C        BLOCK DATA IS REQUIRED FOR VARIABLE GRID SQUARE SIZE APPLICATIO
C
C        LOGICAL UNITS REQUIRED
C
C                 TAPE 1 = PUNCHED OUTPUT
C                 TAPE 3 = PREDAT OUTPUT / METMOD INPUT
C                 TAPE 4 = SETPLT SCRATCH FILE
C                 TAPE 5 = INPUT (TO PREDAT)
C                 TAPE 6 = PRINTED OUTPUT
C                 TAPES 11-20 = POSSIBLE WIND DATA SCRATCH FILES
C                 LTAPE = METEOROLOGICAL DATA FILE (USE 3 FOR CARDS)
C
C
      COMMON /AIRQAL/ NEEDAQ,   NOSPEC,   CONAM(10),   CON(24,25,10)
      COMMON /CNTROL/ KSTOP,             TSUN
      COMMON /DATES/  IDATES(10),  ICT(10),  NFILES
      COMMON  /GRID/  X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT
      COMMON/INPUTS/  TITLE(20),         JDATE(10),          NCURV
      COMMON /KZINPT/ NEEDKZ,            KZPRTX,             NUMSD,
     1                TIMESD(3),        .TEMPSD(50,3),       ZELVSD(50,3)
     2         ,      TEMPSF(24),        NPTSD(3)      ,     NKZDAT
      COMMON /ORIGIN/ UTMXOR, UTMYOR
      COMMON /REUSE/  KPSTAT,  KPWDAT,  CONVRT,  KXTRA, KPWIND
      COMMON /TRAJ/   TSTART,              P(2,100),            ID(2,100)
      COMMON /WDATA/  NUMSTA,  SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      COMMON /WHERE/  RLAT,              RLONG,              THZONE
      COMMON /WIND/   T(100),              V(100),              TH(100),
     1                NPTS
      COMMON /WINFLD/ AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)
      EQUIVALENCE  (RMOR,MORE)
      DATA YES,NEG,MORE,END/3HYES,2HNO,4HMORE,3HEND/
      DATA DALITE/4HDAYL/
      DATA RTD/57.29578/
      DATA  LIN, LOUT /3,6/
      IMAS = 0
      NCURV = 0
      TERM = END
C        CREATE DATA CARD IMAGES ....
100   CALL PREDAT
C          TITLE   --  ANY USER SPECIFIED TITLE
C          TRAJEX  --  DID YOU INPUT A TRAJECTORY (T,V,+TH) EXTERNALLY?
```

```
C        KPWIND  --  DO YOU WANT THE OUTPUTS PUNCHED?
C        NEEDAQ  --  DO YOU WANT AIR QUALITY DATA INTERPOLATED
C                    ALONG THE TRAJECTORY?
C        NEEDKZ  --  VERTICAL DIFFUSITY CALCULATION DESIRED (YES OR NO)
C        LTAPE   --  LOGICAL UNIT OF WIND/MET DATA PERIPHERAL.
C
      READ(LIN,25) TITLE

      READ(LIN,29) TRAJEX
      READ(LIN,29) KPWIND
      READ(LIN,29) NEEDAQ
      READ(LIN,29) NEEDKZ
      READ(LIN,28) LTAPE
C
      CALL MDATE(JDATE)
      IF (TRAJEX.EQ.YES) GO TO 120
      CALL WINDY (IWAS,LTAPE)
      IWAS = 1
      IF (KSTOP.EQ.1) GO TO 810
      GO TO 150
120   READ(LIN,3) P(1,1),P(2,1)
      READ(LIN,29) STIME
      TSUN = 0.
      IF (STIME.EQ.DALITE) TSUN = 0100.
      DO 130 I = 1,101
      READ(LIN,3) A,B,C
      IF (A.LT.0.) GO TO 140
      T(I) = A
      V(I) = B
      TH(I) = C/RTD
130   CONTINUE
      WRITE(LOUT,2)
      GO TO 810
140   NPTS = I-1
      CALL TRAJEC(IWAS,LTAPE)
      IWAS = 1
C
150   CONTINUE
C
810   READ(LIN,25) TERM
      IF (TERM.EQ.RMOR) GO TO 100
      STOP
C
2     FORMAT (1H1,10X,45HDIMENSIONS OF ARRAYS T, V, TH IN COMMON/WIND/
     133H HAVE BEEN EXCEEDED. JOB ABORTED.)
3     FORMAT (40X,3F10.3)
25    FORMAT (20A4)
28    FORMAT (40X,I2)
29    FORMAT (40X,A4)
      END
      SUBROUTINE ANGTST(XS,YS,NGOOD,KUSE,OK)
C
C THIS ROUTINE PERFORMS THE ANGLE TEST
C
C XS     X-COORDINATE UF POSITION
C YS     Y-COORDINATE OF POSITION
C NGOOD  NUMBER OF POINTS-1
C KUSE   INDEX OF CLOSEST STATION
C OK     FLAG IF POSITIVE ANGLE TEST IS OK OTHERWISE LOOK FOR NEW POINT
C
      COMMON /NDATA/ NUMSTA, SSTAN(2,25), ISTANS(2,25), RMIN, RMAX
C
      DIMENSION RVEC(3,3),SVEC(3,3)
      DATA COSTST/.99939083/
C
      OK = -1.
      NUSE = NGOOD+1
      GO TO (100,120,130),NUSE
C
C STORE FIRST VECTOR

100   CONTINUE
      RVEC(1,1) = XS-SSTAN(1,KUSE)
      RVEC(2,1) = YS-SSTAN(2,KUSE)
      RVEC(3,1) = 0.0
110   CONTINUE
      OK = 1.0
      RETURN
C
C STORE SECOND VECTOR AND CHECK ANGLE
120   CONTINUE
      RVEC(1,2) = XS-SSTAN(1,KUSE)
      RVEC(2,2) = YS-SSTAN(2,KUSE)
      RVEC(3,2) = 0.0
      CALL UNITV(RVEC(1,1),SVEC(1,1))
      CALL UNITV(RVEC(1,2),SVEC(1,2))
      COSEY = DOTTY(SVEC(1,1),SVEC(1,2))
C
C     PROVIDE A 2 DEGREE ANGLE CHECK
      IF (COSEY.GT.COSTST) RETURN
      GO TO 110
C
C STORE THIRD VECTOR AND CHECK ANGLE
130   CONTINUE
      RVEC(1,3) = XS-SSTAN(1,KUSE)
      RVEC(2,3) = YS-SSTAN(2,KUSE)
      RVEC(3,3) = 0.0
      CALL UNITV(RVEC(1,3),SVEC(1,3))
      COSEY = DOTTY(SVEC(1,1),SVEC(1,3))
      IF (COSEY.GT.COSTST) RETURN
      COSEY = DOTTY(SVEC(1,2),SVEC(1,3))
      IF (COSEY.GT.COSTST) RETURN
      GO TO 110
C
      END
      SUBROUTINE AZVDIS(NP,AZV,DDD,AZM,VEL,IRF)
C
C THIS SUBROUTINE TAKES THE RAW AZIMUTH AND VELOCITY DATA AND
C PRODUCES ONE AZIMUTH AND VELOCITY
C
C     NP          NUMBER OF RAW DATA POINTS   (UP TO THREE)
C     AZV(1)      AZIMUTH IN WIND PT COORDINATES
C     AZV(2)      VELOCITY IN MPH
C     AZV(3)      NUMBER OF POINTS IN DIGITAL WIND SYSTEM
C     AZV(4)      STATION NAME
C     DDD         DISTANCE FROM INPUT PT TO CLOSEST STATIONS
C     AZM         AZIMUTH TO BE CALCULATED
C     VEL         VELOCITY TO BE CALCULATED
C     IRF         INVERSE OR INVERSE SQUARE DISTANCE CALCULATION TO BE
C
      DIMENSION AZV(12),DDD(3)
      DIMENSION AZT(3),W(3),R(3),AX(3),AY(3)
      COMMON /ADATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      DATA RPD/.01745329/,DPR/57.2957795/
C
      SUM = 0.0
      DO 120 I = 1,NP
      L = I-1
      APT = AZV(4*L+3)
      AZUS = AZV(4*L+1)
      IF (AZUS.GT.APT) AZUS = APT
C
C     COMPUTE SECTOR SIZE

C
      SDEG = 360./APT
C
C     CALCULATE WIND AZIMUTH IN DEGREES CLOCKWISE FROM NORTH
C
      AZT(I) = 180.+AZUS*SDEG
      IF (AZT(I).GT.360.) AZT(I) = AZT(I)-360.
      AX(I) = SIN(AZT(I)*RPD)
      AY(I) = COS(AZT(I)*RPD)
C
C     CALCULATE R(I)
      IF (IRF.EQ.2) GO TO 100
C
C     INVERSE DISTANCE LAW
C
      R(I) = 1./DDD(I)
      GO TO 110
C
C     INVERSE SQUARE DISTANCE LAW
C
100   CONTINUE
      R(I) = 1./(DDD(I)**2)
110   CONTINUE
      SUM = SUM+R(I)
120   CONTINUE
C
```

```
C     CALCULATE W(I) AND THE AZIMUTH AND VELOCITY OF INTEREST
C
      SUMX = 0.
      SUMY = 0.
      SUM2 = 0.0
      DO 130 I = 1,NP
      L = I-1
      W(I) = R(I)/SUM
      SUMX = SUMX+AX(I)*W(I)
      SUMY = SUMY+AY(I)*W(I)
      SUM2 = SUM2+AZV(4*L+2)*W(I)
130   CONTINUE
      AZM = DPR*ATAN2(SUMX,SUMY)
      IF (AZM.LT.0.) AZM = AZM+360.
      VEL = SUM2
      RETURN
C
      END



      SUBROUTINE BARIER(X,Y,XS,YS,OK)
C
C THIS ROUTINE CALCULATES WHAT SIDE OF A BARRIER A POINT IS ON
C BARRIERS ARE FINITE LENGTH LINE SEGMENTS
C
C BAS(1,I)     UTM-X COORDINATE OF FIRST END POINT
C BAS(2,I)     UTM-Y COORDINATE OF FIRST END POINT
C BAS(3,I)     UTM-X COORDINATE OF SECOND END POINT
C BAS(4,I)     UTM-Y COORDINATE OF SECOND END POINT
C NUMBAS       NUMBER OF BARRIERS (UP TO 20)
C IFIN         IF 0 FIRST TIME IN ROUTINE SETUP PT SLOPE LINES
C SLPIN(1,I)   SLOPE OF LINE
C SLPIN(2,I)   INTERCEPT OF LINE
C X            X COORDINATE OF POINT OF INTEREST
C Y            Y COORDINATE OF POINT OF INTEREST
C XS           X COORDINATE OF REFERENCE POINT
C YS           Y COORDINATE OF REFERENCE POINT
C OK           FLAG IF POS (X-Y) IS ON SAME SIDE AS (XS-YS)

C
      DIMENSION BAS(4,20),SLPIN(2,20)
      DIMENSION A(3), B(3), C(3)
      COMMON  /ORIGIN/  UTMXOR, UTMYOR
      DATA IFIN/0/
C
C     ----------------------- NOTE ------------------------------------
C          THE USER MUST SET THE NUMBER OF BARRIERS AND THE  BARRIER
C          END POINT COORDINATES (UTM) IN DATA STATEMENTS BELOW.
C
C          THERE ARE NO SIGNIFICANT GEOGRAPHIC BARRIERS TO THE WIND
C          FIELD IN ST. LOUIS.  NUMBAS AND BAS HAVE BEEN SET TO ZERO.
C
      DATA NUMBAS/0/
      DATA BAS/80*0./
```

```
C
      OK = 1.0
      IF (NUMBAS.LE.0) GO TO 140
      IF (IFIN.GT.0) GO TO 110
C
C SETUP SLOPE INTERCEPT FORM OF A LINE
      DO 100 I = 1,NUMBAS
      DY = BAS(2,I)-BAS(4,I)
      DX = BAS(1,I)-BAS(3,I)
      SLPIN(1,I) = DY/DX
      SLPIN(2,I) = BAS(2,I)-SLPIN(1,I)*BAS(1,I)
100   CONTINUE
      IFIN = 1
C
110   CONTINUE
C
      X = X + UTMXOR
      Y = Y + UTMYOR
      XS = XS + UTMXOR
      YS = YS + UTMYOR
C
      C(1) = XS-X
      C(2) = YS-Y
      A(3) = 0.
      B(3) = 0.
      C(3) = 0.
      DO 120 I = 1,NUMBAS
      D1 = YS-SLPIN(1,I)*XS-SLPIN(2,I)
      D2 = Y-SLPIN(1,I)*X-SLPIN(2,I)
      IF (D1*D2.GE.0.) GO TO 120
      A(1) = BAS(1,I)-X
      A(2) = BAS(2,I)-Y
      B(1) = BAS(3,I)-X
      B(2) = BAS(4,I)-Y
      COSAB = UNIDOT(A,B)
      COSAC = UNIDOT(A,C)
      COSBC = UNIDOT(B,C)
      IF (COSAC.LT.COSAB.OR.COSBC.LT.COSAB) GO TO 120
      OK = -1.0
      GO TO 130
120   CONTINUE
130   CONTINUE
C
      X = X - UTMXOR
      Y = Y - UTMYOR
      XS = XS - UTMXOR
      YS = YS - UTMYOR

C
140   CONTINUE
      RETURN
      END
      SUBROUTINE CELAVG(DH,NMESHP,CELVAT,AKZ,NZHI,STORE)
C
C     *CELAVG* CALCULATES THE AVERAGE DIFFUSIVITY COEFFICIENTS
C              BETWEEN VERTICAL MESH POINT ELEVATIONS FOR USE
C              IN THE CHEMICAL-DIFFUSION MODULE.
C
      DIMENSION CELVAT(1),AKZ(1),STORE(6)
      NP1=NMESHP+1
      DO 20 K=2,NMESHP
      NTOP=IFIX(CELVAT(K)/DH+.5)
      NBOT=IFIX(CELVAT(K-1)/DH+.5)+1
      IF(K.EQ.2) NBOT=2
      HTOP=DH*(NTOP-1)
      HBOT=DH*(NBOT-1)
      SUM=0.
      KSUM=0
      IF(NTOP.GE.NZHI) NTOP=NZHI
      IF(NBOT.GE.NZHI) NBOT=NZHI-1
C
      DO 15 J=NBOT,NTOP
      SUM=SUM+AKZ(J)
      KSUM=KSUM+1
   15 CONTINUE
      AVGKZ=SUM/KSUM
      STORE(K)=AVGKZ
   20 CONTINUE
      STORE(1) = STORE(2)
      STORE(NP1) = STORE(NMESHP)
      RETURN
      END
```

```
      FUNCTION CLOCKT(TSTART,DT)
C
C         CLOCKT RETURNS MILITARY CLOCK TIME AT TSTART + DT
C              TSTART = INITIAL CLOCK TIME
C                  DT = ELAPSED TIME IN MINUTES
C
      XMIN = AMOD(TSTART,100.)
      HRS = TSTART-XMIN
      XMIN = XMIN+DT
      RMIN = AMOD(XMIN,60.)
      HRS = HRS+100.*(XMIN-RMIN)/60.
      CLOCKT = HRS+RMIN
      IF (CLOCKT.GT.2400.) CLOCKT = AMOD(CLOCKT,2400.)
      RETURN
      END
      SUBROUTINE CROSIT
C
C         CROSIT GENERATES TRAJECTORY GRID SQUARE ID HISTORY
C
      DIMENSION ICHAR(26),  TX(4), KSYM(50), IDVG(501)
      COMMON /CNTROL/ KSTOP, TSUN
      COMMON /REUSE/  KPSTAT, KPWDAT, CONVRT, KXTRA, KPWIND
      COMMON /CROSS/ TC(500), PC(2,500), IDC(2,500), NC
      COMMON /GRID/ X1,X2,Y1,Y2,NX,NY,DELX,DELY,DELT
      COMMON /INPUTS/  TITLE(20),  JDATE(10),  NCURV
      COMMON /ORIGIN/  UTMXOR,  UTMYOR
      COMMON /TRAJ/ TSTART, P(2,100), ID(2,100)
      COMMON /WIND/ T(100), V(100), TH(100), NPTS
      COMMON /VGRID/  ITGRID,   IVG(100,100)

C
      DATA    ICHAR/1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1HL,1HM,
     1             1HN,1HO,1HP,1HQ,1HR,1HS,1HT,1HU,1HV,1HW,1HX,1HY,1HZ/
      DATA NCMAX/500/, JTMAX/500/
      DATA SRD/57.29577951/
      DATA NONO/4HNO  /
      DATA KYES /4HYES /
      DATA LOUT, LPUNCH, IDFALT /6,1,0/
C
      KSTOP = 0
      CALL SETTUP
      IF (NPTS.GE.2) GO TO 100
      KSTOP = 1
      WRITE(LOUT,5)
      RETURN
100   CONTINUE
C
C         TABULATE ALL TIMES ASSOCIATED WITH GRID LINE CROSSINGS
C
      NC = 0
      KSYM(1) = ICHAR(1)
      CALL PLTPNT(P(1,1),P(2,1),KSYM(1))
      DO 250 K = 2,NPTS
      L = K-1
      DELT = T(K)-T(L)
      VX = V(L)*COS(TH(L))
      VY = V(L)*SIN(TH(L))
      CALL PLACIT(DELT,L,VX,VY,P(1,K),ID(1,K))
      TSAVE = 0.
      TXMAX = 9999.
      IF (VX.NE.0.) TXMAX = ABS(DELX/VX)
      TYMAX = 9999.
      IF (VY.NE.0.) TYMAX = ABS(DELY/VY)
      RX = P(1,L)-(ID(1,L)-1)*DELX
      RY = P(2,L)-(ID(2,L)-1)*DELY
110   IF (VX) 120,130,140
120   TX(1) = -RX/VX
      GO TO 150
130   TX(1) = 9999.
      GO TO 150
```

```
140   TX(1) = (DELX-RX)/VX
150   CONTINUE
      IF (VY) 160,170,180
160   TY = -RY/VY
      GO TO 190
170   TY = 9999.
      GO TO 190
180   TY = (DELY-RY)/VY
190   CONTINUE
      TS = AMIN1(TX(1),TY) + TSAVE
      TB = AMAX1(TX(1),TY) + TSAVE
      IF (TS.GT.DELT) GO TO 240
      TNXT = TS
      TSNX = AMIN1(TXMAX,TYMAX)
200   CONTINUE
      NC = NC+1
      TC(NC) = T(L)+TNXT
      CALL PLACIT(TNXT,L,VX,VY,PC(1,NC),IDC(1,NC))
      IF (NC.EQ.NCMAX) GO TO 260
      TSAVE = TNXT
      TNXT = TNXT+TSNX
      IF (TNXT-DELT) 210,210,220

210   IF (TNXT-TB) 200,200,230
220   IF (DELT-TB) 240,230,230
230   CONTINUE
      NC = NC+1
      TC(NC) = T(L)+TB
      CALL PLACIT(TB,L,VX,VY,PC(1,NC),IDC(1,NC))
      IF (NC.EQ.NCMAX) GO TO 260
      TSAVE = TB
      RX = PC(1,NC)-(IDC(1,NC)-1)*DELX
      RY = PC(2,NC)-(IDC(2,NC)-1)*DELY
      GO TO 110
240   CONTINUE
      KK = K
      IF (K.GT.26) KK = MOD(K,26)
      IF (K.EQ.52) KK = 26
      KSYM(K) = ICHAR(KK)
      CALL PLTPNT(P(1,K),P(2,K),KSYM(K))
250   CONTINUE
      GO TO 270
260   KSTOP = 2
      NPTS = K
270   CONTINUE
      K = NPTS
C
C         PRINT GRID LOCATIONS AND WIND VECTORS AT TRAJECTORY NODES
C
      STRT = TIMIN(TSTART)
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,1)
      DO 290 L = 1,K
      IF (MOD(L,25).NE.0) GO TO 280
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,1)
  280 CONTINUE
      VV = V(L)*60.
      ANG = TH(L)*SRD
      T(L) = T(L) + STRT
      WRITE(LOUT,2) KSYM(L),T(L),ID(1,L),ID(2,L),P(1,L),P(2,L),VV,ANG
  290 CONTINUE
C
      CALL PLTOUT(LOUT)
C
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,3)
      P1 = P(1,1) + UTMXOR
      P2 = P(2,1) + UTMYOR
      WRITE(LOUT,4)  STRT, P1, P2, ID(1,1), ID(2,1)
      IF(ITGRID.NE.KYES) WRITE(LPUNCH,9)  STRT,P1,P2,ID(1,1),ID(2,1)
      DO 310 L = 1,NC
      IF (MOD(L,25).NE.0) GO TO 300
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,3)
  300 CONTINUE
      TC(L) = TC(L) + STRT
      PC(1,L) = PC(1,L) + UTMXOR
      PC(2,L) = PC(2,L) + UTMYOR
      WRITE(LOUT,4) TC(L),PC(1,L),PC(2,L),IDC(1,L),IDC(2,L)
      IF(ITGRID.EQ.KYES)  GO TO 310
      IF(KPWIND.EQ.KYES)
     1  WRITE (LPUNCH,9) TC(L),PC(1,L),PC(2,L),IDC(1,L),IDC(2,L)
  310 CONTINUE
      A = -10.

      IF(ITGRID.NE.KYES) WRITE (7,9) A
C
      IF (KSTOP.EQ.2) WRITE(LOUT,6)
      KSTOP = 0
C
C         GENERATE GRID SQUARE ID HISTORY FOR VARIABLE SIZE GRID
C
      IF(ITGRID.NE.KYES)  GO TO 360
  320 CONTINUE
      I = ID(1,1)
      J = ID(2,1)
      IF(I.GE.1  .AND. I.LE.NX .AND. J.GE.1 .AND. J.LE.NY)  GO TO 322
      IDVG(1) = IDFALT
      GO TO 323
  322 IDVG(1) = IVG(I,J)
  323 IOK = KYES
      DO 330 K=1,NC
      I = IDC(1,K)
      J = IDC(2,K)
      IOK = KYES
      IF(I.LT.1)  IOK = NONO
      IF(I.GT.NX) IOK = NONO
      IF(J.LT.1)  IOK = NONO
      IF(J.GT.NY) IOK = NONO
      IF(IOK.EQ.KYES)  GO TO 325
      IDVG(K+1) = IDFALT                                                MM   530
      GO TO 330                                                         MM   531
  325 CONTINUE                                                          MM   532
      IDVG(K+1) = IVG(I,J)                                              MM   533
  330 CONTINUE                                                          MM   534
      CALL NEWPAG(TITLE,0,JDATE)                                        MM   535
      WRITE(LOUT,12)                                                    MM   536
      WRITE(LOUT,13) STRT,P1,P2,IDVG(1)                                 MM   537
      IF(KPWIND.EQ.KYES)  WRITE (LPUNCH,10)  TITLE                      MM   538
      IF(KPWIND.EQ.KYES)  WRITE(LPUNCH,14)   STRT, P1, P2, IDVG(1)      MM   539
      N = 1                                                             MM   540
      NLST = N                                                          MM   541
      DO 350 K = 1,NC                                                   MM   542
      IF(MOD(N,25).NE.0)  GO TO 340                                     MM   543
      IF(N.EQ.NLST)  GO TO 340                                          MM   544
      CALL NEWPAG(TITLE,0,JDATE)                                        MM   545
      WRITE(LOUT,12)                                                    MM   546
      NLST = N                                                          MM   547
  340 CONTINUE                                                          MM   548
      IF(IDVG(K+1) .EQ. IDVG(K))  GO TO 350                             MM   549
      N = N + 1                                                         MM   550
      WRITE(LOUT,13)  TC(K), PC(1,K), PC(2,K), IDVG(K+1)                MM   551
      IF(KPWIND.EQ.KYES) WRITE(LPUNCH,14)TC(K),PC(1,K),PC(2,K),IDVG(K+1)MM   552
  350 CONTINUE                                                          MM   553
      IF(KPWIND.EQ.KYES)  WRITE(LPUNCH,14) A                            MM   554
  360 CONTINUE                                                          MM   555
      RETURN                                                            MM   556
C                                                                       MM   557
C                                                                       MM   558
1      FORMAT (1H0,15HTRAJECTORY DATA,/                                 MM   559
     1  1H0,48H  SYMBOL  T(MIN)      I      J      X(KM)      Y(KM),    MM   560
     2  3X,17HV(KM/HR)     THETA  )                                     MM   561
2      FORMAT (/6X,A1,F10.2,2I6,4F10.2)                                 MM   562
3      FORMAT (1H0,14X,3HUTM,7X,3HUTM,/,                                MM   563
     1             3X,7HTC(MIN),4X,6HXC(KM),4X,6HYC(KM),4X,2HIC,4X,2HJC ) MM   564
4      FORMAT (/3F10.2,2I6)                                             MM   565
5      FORMAT (1H0,63HSUBROUTINE CROSIT HAS RECEIVED LESS THAN  2  TRAJECMM   566
                                                                        MM   567
     1TORY POINTS. / 17H CASE TERMINATED. //)                           MM   568
6      FORMAT (1H0,70HSUBROUTINE CROSIT TERMINATED TRAJECTORY EARLY TO AVMM   569
     1OID ARRAY OVERFLOW. //)                                           MM   570
  9    FORMAT (10X,3F10.2,2I10)                                         MM   571
  10   FORMAT (20A4)                                                    MM   572
  12   FORMAT (1H0,2X,10HTIME (MIN) ,5X,20H   UTM-X      UTM-Y    ,     MM   573
     1  5X,15HGRID SQUARE ID     )                                      MM   574
  13   FORMAT (1H0,3X,F7.2,6X,2F10.2,5X,I10)                            MM   575
  14   FORMAT (10X,3F10.2, I10)                                         MM   576
       END                                                              MM   577
                                                                        MM   578
```

CORRECTIONS

1) Before MM 554 insert

```
ITLAST = IDVG (k+1)
TTLAST = TC (K)
```

2) After MM 554 insert

```
    IF(TTLAST.EQ.T(NPTS) GO TO 355
    XX = P(1,NPTS) + UTMXOR
    YY= P(2.NPTS) + UTMYOR
    TI= T(NPTS)
    WRITE(LOUT,13) TI,XX,YY,ITLAST
    IF(KPWIND.EQ.KYES) WRITE(LPUNCH,14) TI,XX,YY,ITLAST
355 CONTINUE
```

```
      SUBROUTINE DATE(IOLD,MOVE,IWAY,NEW)
C
C     THIS SUBROUTINE CHANGES THE TIME AND DATE
C
      DIMENSION IOLD(2),NEW(2)
C
C     IOLD(1)  OLD TIME 24 HOUR CLOCK
C     IOLD(2)  OLD DATE YEAR-MONTH-DAY
C     MOVE     TIME INCREMENT
C     IWAY     POSITIVE OR NEGATIVE APPLICATION OF THE INCREMENT
C     NEW(1)   NEW TIME 24 HOUR CLOCK
C     NEW(2)   NEW DATE YEAR-MONTH-DAY
C
      I1 = IOLD(1)/100
      I2 = IOLD(1)-100*I1
      J1 = IOLD(2)/10000
      J2 = (IOLD(2)-10000*J1)/100
      J3 = IOLD(2)-10000*J1-100*J2
      NEW(2) = IOLD(2)
      K22 = I1
      IF (IWAY.LT.0) GO TO 150
C
C     INCREMENT THE TIME
      K2 = I2+MOVE
      IF (K2.LT.60) GO TO 140
      K2 = K2-60
      K22 = I1+1
      IF (K22.LE.24) GO TO 140
      K22 = K22-24
      NEW(1) = 100*K22+K2
      K22 = J2
      K2 = J3+1
      IF (J2.EQ.2) GO TO 120
      IF (J2.EQ.4) GO TO 130
      IF (J2.EQ.6) GO TO 130
      IF (J2.EQ.9) GO TO 130
      IF (J2.EQ.11) GO TO 130
C     31 DAY MONTHS
      IF (K2.LE.31) GO TO 110
      K2 = K2-31
100   CONTINUE
      K22 = J2+1
      IF (K22.LE.12) GO TO 110
      K22 = K22-12
      J1 = J1+1
110   CONTINUE
      NEW(2) = 10000*J1+100*K22+K2
      RETURN
120   CONTINUE
      IF (K2.LE.28) GO TO 110
      K2 = K2-28
      GO TO 100

130   CONTINUE
      IF (K2.LE.30) GO TO 110
      K2 = K2-30
      GO TO 100
 140  CONTINUE
      NEW(1) = 100*K22+K2
      RETURN
C
C     DECREMENT TIME
 150  CONTINUE
      K2 = I2-MOVE
      IF (K2.GE.0) GO TO 140
      K2 = K2+60
      K22 = I1-1
      IF (K22.GE.0) GO TO 140
      K22 = K22+24
      NEW(1) = 100*K22+K2
      K22 = J2
      K2 = J3-1
      IF (K2.GT.0) GO TO 110
      K22 = J2-1
      IF (J2.EQ.12) GO TO 170
      IF (J2.EQ.10) GO TO 170
      IF (J2.EQ.7) GO TO 170
      IF (J2.EQ.5) GO TO 170
      IF (J2.EQ.3) GO TO 180
      K2 = 31
 160  CONTINUE
      IF (K22.GT.0) GO TO 110
      K22 = 12
      J1 = J1-1
      GO TO 110
 170  CONTINUE
      K2 = 30
      GO TO 160
 180  CONTINUE
      K2 = 28
      GO TO 160
C
      END
```

```
C
      SUBROUTINE DIFFUS(STAB,ZHT,AKZ,USTAR)
C
C-----THIS SUBROUTINE CALCULATES THE VALUES OF KZ IN THE SURFACE
C     LAYER USING THE FORMULATIONS OF BUSINGER,ET AL(1971).
C
      AKN=.35*USTAR*ZHT
   66 IF(STAB) 67,68,69
   67 AKZ=AKN*((1.-15.*ZHT/STAB)**0.25)
      GO TO 70
   69 AKZ=AKN/(1.+4.7*ZHT/STAB)
      GO TO 70
   68 CALL EXIT
   70 RETURN
```

```
      END
```

```
      FUNCTION DOTTY(A,B)
C
C RETURNS THE DOT PRODUCT OF VECTOR A AND VECTOR B
C
      DIMENSION A(3),B(3)
C
      DOTTY = 0.0

      DO 100 I = 1,3
      DOTTY = A(I)*B(I)+DOTTY
100   CONTINUE
      RETURN
      END
```

```
      FUNCTION DTIME(A)
C-----THIS FUNCTION CONVERTS TIME ON FROM HOURS AND MINUTES
C     TO HOURS AND DECIMAL FRACTIONS OF AN HOUR.  FOR EXAMPLE,
C     1130 IS CONVERTED TO 11.5.
C
      TIMEX=A*0.01
      DTIME=(TIMEX-AINT(TIMEX))/0.6+AINT(TIMEX)
      RETURN
      END
```

```
      SUBROUTINE EDDY  (KPWIND)
C
C     THIS SUBROUTINE IS THE DRIVER FOR THE VERTICAL EDDY DIFFUSIVITY
C     SUB-MODULE.
C     D. GODDEN  1.25.78
C
      DIMENSION   STORE(6), ZTIME(15),TEMP(201),TTMP(201),IKLAS(15)
      COMMON /INPUTS/ TITLE(20),        JDATE(10),          NCURV
      COMMON /KZINPT/ NEEDKZ,           KZPRTX,             NUMSD,
     1                TIMESD(3),        TEMPSD(50,3),       ZELVSD(50,3)
     2    ,           TEMPSF(24),       NPTSD(3),           NKZDAT
      COMMON/CELECT/CELHTS(5,1),NOHTS,NMES
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)
      COMMON/DIFDAT/DELTAT(30),STAB(30),ZMIX(30),USTAR(30),
     +             UZZ(201),AKZ(201)
      COMMON/WIND/TIME(100),V(100),TH(100),NPTS
      COMMON/ORIGIN/UTMXOR,UTMYOR
      COMMON/TRAJ/TSTART,P(2,100),ID(2,100)
      COMMON/PEVE/PE(2,24),VE(24)
      COMMON/CORIOL/F
      COMMON/WHERE/STALAT,RLONG,TMZONE
      DATA IYES/4HYES /
      DATA LOUT,LPUNCH  /6,1/
      DATA ADVECT/0./
      DATA ZW/30./
      SUPER = -0.98
      ZSUP = 0.0
      SUP = 0.0
      CALL KZDATA(IERR)
      IF(IERR.LT.0) GO TO 2000
      IDBUG = IERR
      IKKL = 1
      CALL UXYPOS
      DO 1000 ISD=1,NUMSD
      ISDP1 = ISD + 1
      IF(ISD.EQ.NUMSD) GO TO 10
      SUP=(TEMPSD(2,ISDP1)-TEMPSD(1,ISDP1))/(ZELVSD(2,ISDP1)-ZELVSD(1,
     1ISDP1))*100.
      IF(SUP.GE.SUPER) GO TO 10
      SUPER=SUP
      ZSUP=ZELVSD(2,ISDP1)-ZELVSD(1,ISDP1)
   10 IT=1
      ZTIME(IT)=TIMESD(ISD)
      IF(STALAT.EQ.0.) STALAT=35.
      RADLAT=STALAT*0.017453
      F=1.4584E-04*SIN(RADLAT)
      DH=0.
      AKMIN=6.

      ZTIME(IT)=DTIME(ZTIME(IT))
      IN=IFIX(ZTIME(IT))+1
      IF(ZW.EQ.0.) ZW=10.
    2 CALL XMIT(NPTSD(ISD),ZELVSD(1,ISD),Z(1))
      CALL XMIT(NPTSD(ISD),TEMPSD(1,ISD),T(1))
      LMAX=NPTSD(ISD)
      TEMP(1)=T(1)
      T1ST=TEMP(1)
      UZZ(1)=0.
      TMAX=TEMP(1)
      ZMAX=0.
      STAELV=Z(1)
      DO 5 I=1,LMAX
      Z(I)=Z(I)-STAELV
    5 CONTINUE
      LMAXM1=LMAX-1
    6 CALL XMIT(LMAXM1,Z(2),ZZ(1))
      CALL XMIT(LMAXM1,T(2),TT(1))
      ZHIGH=ZZ(LMAXM1)
      TLAPS=(TT(1)-T(1))*100./ZZ(1)
   20 DH=DH+10.
      NMAX=IFIX(ZHIGH/DH+0.5)+1
      IF(NMAX.GT.201) GO TO 20
      DH01=DH*0.01
      L=1
      M=1
    9 IF(L.EQ.LMAX) GO TO 24
      ZZ(M)=AINT(ZZ(L)/DH+0.5)*DH
      IF(ZZ(M).EQ.Z(M)) GO TO 16
      Z(M+1)=ZZ(M)
      T(M+1)=TT(L)
      L=L+1
      M=M+1
      GO TO 9
   16 L=L+1
      GO TO 9
   24 LMAX=M
      LMAXM1=LMAX-1
      CALL XMIT(LMAXM1,Z(2),ZZ(1))
      CALL XMIT(LMAXM1,T(2),TT(1))
      IF(ZSUP.EQ.0.) ZSUP=Z(2)
      CALL SMOOTH(LMAX)
      IF(LMAX.GT.31)  LMAX = 31
      ZHIGH = Z(LMAX)
      IF(ZNITE.EQ.0.) ZNITE=Z(2)
      ZNITE=AMAX1(100.,ZNITE)
    7 DO 12 L=2,LMAX
      TPOT=TT(L-1)+.0098*ZZ(L-1)
      IF(TPOT.LT.TMAX) GO TO 12
      TMAX=TPOT
      ZMAX=Z(L)
   12 CONTINUE
      IF(TLAPS.GE.-.98.OR.ZMAX.GT.0..OR.LMAX.LE.2) GO TO 14
      TMAX=T(2)
      ZMAX=Z(2)
      GO TO 7
   14 N=1
      AKZ(N)=AKMIN
      J=1
    1 DELTAT(J)=(TT(J)-T(J))*100./(ZZ(J)-Z(J))
  107 ZHT=Z(J)
  108 ZHT=ZHT+DH

      N=N+1
      TEMP(N)=TEMP(N-1)+DELTAT(J)*DH01
      IF(ZHT.GE.ZZ(J)) GO TO 8
      GO TO 108
    8 IF(ZZ(J).GE.ZHIGH) GO TO 21
      J=J+1
      GO TO 1
   21 NMAX=N
      IF(TLAPS.GT.-.98) GO TO 33
      N=1
      TTMP(N)=TEMP(N)
      GO TO 330
C-----UPDATE SURFACE TEMPERATURE AND WIND SPEED
   33 N=1
      J=1
      ZHT=0.
      IT=IT+1
      TSTOP=AINT(DTIME(TIME(NPTS)))
      IF(ISD.EQ.NUMSD) GO TO 30
      TSTOP=AINT(DTIME(TIMESD(ISDP1)))
   30 ZTIME(IT)=(AINT(TIME(1)*0.01)+FLOAT(IT-2))
```

```
      IF(ISD.GT.1) ZTIME(IT)=AINT(TIMESD(ISD)*0.01)+FLOAT(IT-1)
      IS=IFIX(ZTIME(IT))+1
      TTMP(1)=TEMPSF(IS)-TEMPSF(IN)+T1ST
      USFC=VE(IS)
      IF(USFC.EQ.0.) USFC=0.5
      TIMEX=ZTIME(IT)-ZTIME(1)
      TTMP(1)=TTMP(1)-(ADVECT*TIMEX)
      CALL RUFNES(Z0,IS)
      IF(KZPRTX.NE.IYES) GO TO 102
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,39) ZTIME(IT)
      IF(IDBUG.EQ.IYES)  WRITE(LOUT,26)
  102 AKZ(1)=AKMIN
      UZZ(1)=0.
      ODZ=0.
      IF(TTMP(1).LT.T(1)) GO TO 40
      IF(TTMP(1).EQ.T(1)) GO TO 35
C-----NEW SFC TEMP GREATER THAN LAST SFC TEMP
  330 DELT=-0.98
   32 N=N+1
      TTMP(N)=TTMP(N-1)+DELT*DH01
      IF(TTMP(N).LT.TEMP(N)) GO TO 34
      IF(N.EQ.NMAX) GO TO 37
      GO TO 32
   34 IF(TTMP(1).LT.TEMP(1)) GO TO 471
      NTRSCT=N-1
      GO TO 36
   37 NTRSCT=NMAX
   36 IF(TTMP(1).LE.TMAX.OR.NTRSCT.LT.NMAX) GO TO 230
      TMAX=TTMP(1)
      ZMAX=0.
  230 ZNVRSN=(NTRSCT-1)*DH
      T(1)=TTMP(1)
      IF(ZNVRSN.EQ.0.) GO TO 249
      IF(ZNVRSN-Z(2)) 231,249,231
  231 DO 239 J=2,LMAX
      IF(J.EQ.2) GO TO 235
  232 IF(ZZ(J-1).LE.ZNVRSN) GO TO 239
      L=L+1
      Z(L)=ZZ(J-1)
      T(L)=TT(J-1)

      GO TO 239
  235 L=J
  236 Z(L)=ZNVRSN
      T(L)=TTMP(NTRSCT)
      IF(Z(L).LT.ZZ(1)) GO TO 232
  239 CONTINUE
      LMAX=L
      DELT=TLAPS
      IF(SUPER.LT.DELT) DELT=SUPER
  249 LMAXM1=LMAX-1
      DO 250 J=1,LMAXM1
      TT(J)=T(J+1)
      ZZ(J)=Z(J+1)
      DELTAT(J)=(TT(J)-T(J))*100./(ZZ(J)-Z(J))
  250 CONTINUE
      IF(DELTAT(1).GT.DELT) DELTAT(1)=DELT
      IF(IT-2) 33,22,23
   22 N1=NTRSCT+1
      NO=NMAX-NTRSCT
      CALL XMIT(NO,TEMP(N1),TTMP(N1))
   23 CALL STADIF(ZW,Z0,USFC,ZHIGH,DH,AKMIN,ISKLAS)
      GO TO 29
C-----NEW SFC TEMP EQUAL TO LAST SFC TEMP
   35 IF(IT.GT.2) GO TO 38
      CALL XMIT(NMAX,TEMP(1),TTMP(1))
   38 CALL STADIF(ZW,Z0,USFC,ZHIGH,DH,AKMIN,ISKLAS)
      GO TO 29
C-----NEW SFC TEMP LESS THAN LAST SFC TEMP
   40 IF(DELTAT(1).LT.-0.98.AND.IT.GT.1) GO TO 470
      T(1)=TTMP(1)
      DELTAT(1)=(T(2)-T(1))*100./Z(2)
      IF(DELTAT(1)+0.98) 45,45,44
   44 IF(Z(2)-ZNITE) 440,45,450
   45 N1=IFIX(ZZ(1)/DH)+1
      DO 400 N=2,N1
      TTMP(N)=TTMP(N-1)+DELTAT(1)*DH01
  400 CONTINUE
      IF(IT.GT.2) GO TO 48
      N11=N1+1
      NO=NMAX-N1
      CALL XMIT(NO,TEMP(N11),TTMP(N11))
   48 CALL STADIF(ZW,Z0,USFC,ZHIGH,DH,AKMIN,ISKLAS)
      GO TO 29
  440 DO 445 J=3,LMAX
      IF(Z(J)-ZNITE) 445,441,442
  441 JSTART=2
      GO TO 443
  442 NITE=IFIX(ZNITE/DH)+1
      T(2)=TEMP(NITE)
      Z(2)=ZNITE
      JSTART=3
  443 NO=LMAX-J+1
      CALL XMIT(NO,ZZ(J-1),Z(JSTART))
      CALL XMIT(NO,TT(J-1),T(JSTART))
      NO=J-JSTART
      JMAX=LMAX-NO+1
      CALL XMIT(-NO,0.,Z(JMAX))
      CALL XMIT(-NO,0.,T(JMAX))
      NO=LMAX-1
      CALL XMIT(NO,Z(2),ZZ(1))
      CALL XMIT(NO,T(2),TT(1))
      LMAX=JMAX-1

      GO TO 446
  445 CONTINUE
  446 JMAX=LMAX-1
      DO 449 I=1,JMAX
      DELTAT(I)=(TT(I)-T(I))*100./(ZZ(I)-Z(I))
  449 CONTINUE
      GO TO 45
  450 T(2)=T(2)+.0098*(Z(2)-ZNITE)
      Z(2)=ZNITE
      DELTAT(1)=(T(2)-T(1))*100./(Z(2)-Z(1))
      DELT=DELTAT(2)
      DELTAT(2)=-.98
      DO 455 M=2,LMAX
      T(M+1)=TT(M-1)
      Z(M+1)=ZZ(M-1)
      STORE=DELTAT(M+1)
      DELTAT(M+1)=DELT
  455 DELT=STORE
      CALL XMIT(LMAX,T(2),TT(1))
      CALL XMIT(LMAX,Z(2),ZZ(1))
      LMAX=LMAX+1
      GO TO 45
  470 T2=T(1)+DELTAT(1)*ZSUP*0.01
  471 DELTAT(1)=(T2-TTMP(1))*100./ZSUP
      IF(DELTAT(1).GT.-0.98) GO TO 40
      T(1)=TTMP(1)
      CALL STADIF(ZW,Z0,USFC,ZHIGH,DH,AKMIN,ISKLAS)
      GO TO 130
   29 CONTINUE
      CALL XMIT(NMAX,TTMP(1),TEMP(1))
  130 CALL KEKLAY(LMAX,ZMAX,ZHIGH,AKMIN,DH)
  290 IF(KZPRTX.NE.IYES) GO TO 301
      WRITE(LOUT,25) DH
      NOPTS=NMAX-1
      WRITE(LOUT,1260)
      NL = NOPTS/5
      NL2 = 2*NL
      NL3 = 3*NL
      NL4 = 4*NL
      NTRUNC = MOD(NOPTS,5)
      DO 100 I=1,NL
      I2 = I+NL
      I3 = I+NL2
      I4 = I+NL3
      I5 = I+NL4
      ZI = DH*FLOAT(I)
      ZI2 = DH*FLOAT(I2)
      ZI3 = DH*FLOAT(I3)
      ZI4 = DH*FLOAT(I4)
      ZI5 = DH*FLOAT(I5)
      WRITE(LOUT,28) ZI,AKZ(I+1),ZI2,AKZ(I2+1),ZI3,AKZ(I3+1),
     1                         ZI4,AKZ(I4+1),ZI5,AKZ(I5+1)
100   CONTINUE
      IF(NTRUNC.EQ.0)  GO TO 95
      I5 = I5 + 1
      DO 94 I =I5,NOPTS
      ZI = DH*FLOAT(I5)
      I5 = I5 + 1
94    WRITE(LOUT,1250) ZI, AKZ(I+1)
95    CONTINUE
```

```
      NOHP1 = NOHTS + 1
      WRITE(LOUT,1190)
      WRITE(LOUT,1210) (IG,IG=1,NOHP1)
      WRITE(LOUT,1220)
  301 IF(IKKL.NE.1)  GO TO 302
      CALL NEWPAG(TITLE,0,IDATE)
      NOHP1 = NOHTS + 1
      WRITE(LOUT,1190)
      WRITE(LOUT,1200)   (CELHTS(IG,1),IG=1,NOHTS)
      WRITE(LOUT,1210)   (IG,IG=1,NOHP1)
      WRITE(LOUT,1220)
302   ITPP = IFIX(ZTIME(IT) + 1)
      CALL CELAVG(DH,NOHTS,CELHTS(1,1),AKZ,NMAX,STORE)
      TJJJ = ZTIME(IT)*100.
      WRITE(LOUT,1230) TJJJ,TEMPSF(ITPP),ISKLAS,(STORE(IG),IG=1,NOHP1)
      TJJJ = .60*TJJJ
      IF(KPWIND.EQ.IYES)
     1   WRITE(LPUNCH,1240) IKKL,TJJJ,(STORE(IG),IG=1,NOHP1)
      IKLAS(IKKL) = ISKLAS
      IKKL = IKKL + 1

      IF(ZTIME(IT).GE.TSTOP) GO TO 1000
      GO TO 33
 1000 CONTINUE
      CALL NEWPAG(TITLE,0,JDATE)
      IKKL = IKKL - 1
      TMINXX = IFIX(TIME(1))*.60
      DO 1005 KK = 1,IKKL
      IF(KPWIND.EQ.IYES) WRITE(LPUNCH,1001) TMINXX, IKLAS(KK)
      TMINXX = TMINXX + 60.0
 1005 CONTINUE
      RETURN
 2000 WRITE(LOUT,2001)
      WRITE(LOUT,2002) IERR
      STOP
   25 FORMAT(1H0,33H VERTICAL EDDY DIFFUSIVITY EVERY ,F3.0,8H METERS  )
   26 FORMAT(1H0,3X,6HHEIGHT,6X,5HDT/DZ,7X,5HDU/DZ,7X,1HA,14X,1HL,8X,
     15HUSTAR,4X,6HMIX HT,/22H   (METERS)   (C/100M),5X,7H(1/SEC),18X,
     28H(METERS),3X,17H(M/SEC)  (METERS))
   28 FORMAT(1H ,5X,5(F6.0,F9.1,5X))
   39 FORMAT(1H0,15X,4HTIME,2X,F4.0,6H00 LST)
 1001 FORMAT(25H TIME AND STABILITY CLASS   ,15X,F10.1,I10)
 1190 FORMAT(1H0,7X,70HAVERAGE DIFFUSION COEFFICIENTS FOR THE CHEMICAL/D
     1IFFUSION MODULE                  /)
 1200 FORMAT(1H0,7X,22HMESH POINT ELEVATIONS ,15X,5F10.0,5X,6HMETERS/ )
 1210 FORMAT(1H0,7X,4HTIME,5X,7HSURFACE,5X,9HSTABILITY,8X,6(2HKZ,I1,7X))
 1220 FORMAT(1H ,14X,11HTEMPERATURE,5X,5HCLASS,30X,11H(M**2/MIN)  )
 1230 FORMAT(1H0,7X,F4.0,6X,F5.1,9X,I2,5X,6F10.0)
 1240 FORMAT(14H DIFFUSIVITIES  ,I2,4X,F10.1,10X,3F10.1,/,20X,3F10.1)
 1250 FORMAT(1H ,85X,F6.0,F9.1)
 1260 FORMAT(1H0,6X,5(6HHEIGHT,5X,2HKZ,7X)/8X5(15H(M)    (M**2/MIN),4X)/)
 2001 FORMAT(1H0,69HERROR IN SUBROUTINE KZDATA WHICH READS IN DATA NEEDE
     1D TO CALCULATE KZ)
 2002 FORMAT(1H ,5HIERR=,I4)
      END
      SUBROUTINE EDGE(X,Y,ITF,IEDGE)
C
C     *EDGE* CHECKS WHETHER NEW POINT IS OUTSIDE THE MODELING REGION
C
      COMMON /GRID/ X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT
      DATA IONE/1/
      DATA BUFZON /15./
C
      XW = X1 - BUFZON
      XE = X2 + BUFZON
      YS = Y1 - BUFZON
      YN = Y2 + BUFZON
C
      IF (X.LT.XW) GO TO 100
      IF (X.GT.XE) GO TO 100
      IF (Y.LT.YS) GO TO 100
      IF (Y.GT.YN) GO TO 100
C
C         POINT IS OK
C
      IEDGE = 0
      GO TO 110
100   CONTINUE
C
C         POINT IS OUTSIDE GRID + BUFFER ZONE
C
      IEDGE = ISIGN(IONE,ITF)
110   RETURN

      END
```

```
      SUBROUTINE EXTRP(IHOUR,ARY,VAL)
C
C      EXTRP SELECTS A VALUE OUT OF A 24-WORD ARRAY.  IF THE VALUE
C      SOUGHT IS MISSING (INDICATED BY A NEGATIVE ENTRY), EXTRP
C      WILL ATTEMPT TO FILL THE GAP BY INTERPOLATION OR EXTRAPOLATION.
C
C     IHOUR       CLOCK TIME ON A 24 HOUR BASIS
C     ARY         ARRAY OF 24 VALUES  (WIND OR VELOCITY)
C     VAL         VALUE SELECTED
C
C     MISSING DATA SITUATIONS HANDLED BY EXTRP
C
C     CASE  I-3  I-2  I-1   I   I+1  I+2  I+3
C      0               X    -    X
C      1          X    X    -    -
C      2     X    -    X    -    -
C      3     X    X    -    -
C      4               -    -    X    X
C      5               -    -    X    -    X
C      6                    -    -    X    X
C
      DIMENSION ARY(24)
```

```
C
      IPICK = IHOUR/100+1
      IF (IPICK.GT.24) IPICK = IPICK-24
      VAL = ARY(IPICK)
      IF (VAL.GE.0.) RETURN
C
C ENTRY MISSING. MUST TRY TO GET VALUE BY INTERPOLATION OR EXTRAPOLATION
C IS THERE A VALUE ON BOTH SIDES
      VAL = -1.
      IF (IPICK.EQ.1) GO TO 100
      IF (IPICK.EQ.24) GO TO 130
      IF (ARY(IPICK-1).LT.0.) GO TO 160
      IF (ARY(IPICK+1).LT.0.) GO TO 190
C
C CASE 0
C
      VAL = 0.5*(ARY(IPICK-1)+ARY(IPICK+1))
      RETURN
C
C WE ARE AT THE FIRST POINT
100   CONTINUE
      IF (ARY(2).LT.0.) GO TO 110
      IF (ARY(3).LT.0.) GO TO 120
      VAL = ARY(2)-(ARY(3)-ARY(2))
      GO TO 220
110   CONTINUE
      IF (ARY(3).LT.0..OR.ARY(4).LT.0.) RETURN
      VAL = ARY(3)-2.*(ARY(4)-ARY(3))
      GO TO 220
120   CONTINUE
      IF (ARY(4).LT.0.) RETURN
      ARY(3) = 0.5*(ARY(2)+ARY(4))
      GO TO 100
C
C WE ARE AT THE LAST POINT
130   CONTINUE
      IF (ARY(23).LT.0.) GO TO 140
      IF (ARY(22).LT.0.) GO TO 150
      VAL = ARY(23)+ARY(23)-ARY(22)

      GO TO 220
140   CONTINUE
      IF (ARY(22).LT.0..OR.ARY(21).LT.0.) RETURN
      VAL = ARY(22)+2.*(ARY(22)-ARY(21))
      GO TO 220
150   CONTINUE
      IF (ARY(21).LT.0.) RETURN
      ARY(22) = 0.5*(ARY(21)+ARY(23))
      GO TO 130
160   CONTINUE
      IF (ARY(IPICK+1).LT.0.) GO TO 180
      IF (IPICK+2.GT.24) GO TO 180
      IF (ARY(IPICK+2).LT.0.) GO TO 170
C
C CASE 4
```

```
      VAL = ARY(IPICK+1)-(ARY(IPICK+2)-ARY(IPICK+1))
      GO TO 220
170   CONTINUE
      IF (IPICK+3.GT.24) GO TO 180
      IF (ARY(IPICK+3).LT.0.) GO TO 180
C
C CASE 5
      VAL = ARY(IPICK+1)-0.5*(ARY(IPICK+3)-ARY(IPICK+1))
      GO TO 220
180   CONTINUE
      VAL = VAL-1.
      IF (VAL.LT.-3.) RETURN
      IF (IPICK-2.LT.1) GO TO 210
      IF (ARY(IPICK-2).LT.0.) GO TO 210
      IF (IPICK-3.LT.1) GO TO 210
      IF (ARY(IPICK-3).LT.0.) GO TO 210
C
C CASE 3
      VAL = ARY(IPICK-2)+2.*(ARY(IPICK-2)-ARY(IPICK-3))
      GO TO 220
190   CONTINUE
      IF (IPICK-2.LT.1) GO TO 210
      IF (ARY(IPICK-2).LT.0.) GO TO 200
C
C CASE 1
      VAL = ARY(IPICK-1)+(ARY(IPICK-1)-ARY(IPICK-2))
      GO TO 220
200   CONTINUE
      IF (IPICK-3.LT.1) GO TO 210
      IF (ARY(IPICK-3).LT.0.) GO TO 210
C
C CASE 2
      VAL = ARY(IPICK-1)+0.5*(ARY(IPICK-1)-ARY(IPICK-3))
      GO TO 220
210   CONTINUE
      VAL = VAL-1.
      IF (VAL.LT.-3.) RETURN
      IF (IPICK+2.GT.24) GO TO 180
      IF (ARY(IPICK+2).LT.0.) GO TO 180
      IF (IPICK+3.GT.24) GO TO 180
      IF (ARY(IPICK+3).LT.0.) GO TO 180
C
C CASE 6
      VAL = ARY(IPICK+2)-2.*(ARY(IPICK+3)-ARY(IPICK+2))
220   CONTINUE
      IF (VAL.LT.0.) VAL = 0.
      RETURN

C
      END
      FUNCTION FULGOL(A,ZO)
C-----THIS FUNCTION COMPUTES 1/L FROM THE FULLE PARAMETER A USING
C     GOLDER'S CONVERSIONS FROM STABILITY CATEGORIES TO VALUES OF
C     1/L.
C
      IF(A.GT.8.0) GO TO 101
      SLOPE=0.0156-0.0072*(A-7.)
      XL=0.0540*(A-8.33)+SLOPE*ALOG(2.*ZO)
      IF(XL.GT.-0.003) XL=-0.003
      GO TO 107
  101 IF(A.GT.8.75) GO TO 102
      SLOPE=0.0084-0.0085*(A-8.)
      XL=0.0213*(A-8.84)+SLOPE*ALOG(2.*ZO)
      IF(XL.GT.-0.0001) XL=-0.0001
      GO TO 107
  102 IF(A.GT.9.02) GO TO 103
      SLOPE=0.0019-0.0070*(A-8.75)
      XL=0.0074*(A-9.02)+SLOPE*ALOG(2.*ZO)
      IF(XL.GT.-0.00001) XL=-0.00001
      GO TO 107
  103 IF(A.GT.9.5) GO TO 104
      SLOPE=-0.0029*(A-9.02)
      XL=0.0031*(A-9.02)+SLOPE*ALOG(2.*ZO)
      IF(XL.LT.0.00001) XL=0.00001
      GO TO 107
  104 IF(A.GT.11.25) GO TO 105
      SLOPE=-0.0014-0.0040*(A-9.5)
      XL=0.0083*(A-9.32)+SLOPE*ALOG(2.*ZO)
      IF(XL.LT.0.0001) XL=0.0001
      GO TO 107
  105 IF(A.GT.12.375) GO TO 106
      SLOPE=-0.0084-0.0069*(A-11.25)
      XL=0.0083*(A-9.32)+SLOPE*ALOG(2.*ZO)
      IF(XL.LT.0.004) XL=0.004
      GO TO 107
  106 SLOPE=-0.0162-0.0069*(A-12.375)
      XL=0.0083*(A-9.32)+SLOPE*ALOG(2.*ZO)
      IF(XL.LT.0.00667) XL=0.00667
  107 FULGOL=XL
      RETURN
      END


      SUBROUTINE GETAZV(KOLD,ID,XS,YS,NC,AZV,NGOOD,DDIS)
C
C     THIS SUBROUTINE RETRIEVES MEASUREMENT RECORDS FROM
C     CLOSE STATIONS.
C
C     KOLD(1)  - TIME - 24 HOUR CLOCK
C     KOLD(2)  DATE - YEAR-MONTH-DAY
C     ID       STATION ID (IF ANY)
C     XS       X COORDINATE OF POSITION
C     YS       Y COORDINATE OF POSITION
C     NC       NUMBER OF CLOSEST STATIONS TO USE
```

```
C     AZV(1)   AZIMUTH
C     AZV(2)   VELOCITY
C     AZV(3)   NUMBER OF DIGITAL WIND POINTS
C     AZV(4)   NAME OF CLOSEST POINT
C     NGOOD    NUMBER OF POINTS ACTUALLY FOUND TO BE GOOD
C     DDIS     DISTANCE TO POINTS FOUND
      DIMENSION KOLD(2),AZV(12)
      DIMENSION PT(24),XIMPH(24)

      DIMENSION DIS(25) ,  IDIS(25)
      DIMENSION DDIS(2)
      DIMENSION ID(2)
      DIMENSION NAM(2)
      DIMENSION NAME(2)
      DIMENSION   NWDATA(52,25)
      COMMON /DATES/  IDATE(10),  ICT(10),  NFILES
      COMMON  /ORIGIN/  UTMXOR, UTMYOR
      COMMON /WDATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      COMMON /WINFLD/  AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)
      EQUIVALENCE   (NWDATA(1,1),AWDATA(1,1))
      EQUIVALENCE (XID,KD)
      EQUIVALENCE  (DIS,DISAVE), (IDIS,IDSAVE)
      DATA IBLK /1H /, IFILE/11/
      DATA   INCOR/0/
      DATA LOUT /6/
C
C ZERO OUT AZV ARRAY
      KD = IBLK
      DO 100 I = 1,3
      L = I-1
      AZV(4*L+1) = 0
      AZV(4*L+2) = 0
      AZV(4*L+3) = XID
      AZV(4*L+4) = XID
100   CONTINUE
C
C************************    ST. LOUIS ONLY   **************************
C
C         REMOVE THE CARDS BETWEEN THE STARED LINES FOR OTHER CITIES
C
C
      I = 1
      IF = 1
      IF(ID(1) .EQ. IBLK  )  GO TO 170
      GO TO 145
  105 CONTINUE
************************************************************************
C     ARE WE AT A STATION
      IF (ID(1).EQ.IBLK) GO TO 170
C
C     YES GET THE STATION DATA
      DO 110 I = 1,NFILES
      IF (KOLD(2).EQ.IDATE(I)) GO TO 130
110   CONTINUE
120   CONTINUE
```

```
      WRITE(LOUT,1) KOLD(2)
      NGOOD = 0
      RETURN
130   CONTINUE
      IC = 0
      IF = IFILE + I
      IF(INCOR.EQ.IDATE(I))  GO TO 145
      REWIND IF
140   CONTINUE
      IC = IC+1
      IF(IC.GT.ICT(I)) GO TO 144
      READ (IF) NWDTE,NAM,APT,PT,XIMPH
      NWDATA(1,IC) = NWDTE
      NWDATA(2,IC) = NAM(1)
      NWDATA(3,IC) = NAM(2)
      AWDATA(4,IC) = APT

      CALL XMIT(24,PT,AWDATA(5,IC))
      CALL XMIT(24,XIMPH,AWDATA(29,IC))
      GO TO 140
C
C
144   INCOR = IDATE(I)
145   CONTINUE
C
C     SEARCH FOR STATION
C
      M = ICT(I)
      DO 148 K = 1,M
      NAM(1) = NWDATA(2,K)
      NAM(2) = NWDATA(3,K)
      IF(ID(1).EQ.NAM(1).AND.ID(2).EQ.NAM(2)) GO TO 149
148   CONTINUE
      GO TO 150
149   CONTINUE
      CALL XMIT(24,AWDATA(5,K),PT)
      CALL XMIT(24,AWDATA(29,K),XIMPH)
      APT = AWDATA(4,K)
C
C     WE HAVE FOUND THE DATA, NOW EXTRAPOLATE IF NECESSARY
      CALL EXTRP(KOLD,PT,AZV(1))
      IF (AZV(1).LT.0.) GO TO 160
      CALL EXTRP(KOLD,XIMPH,AZV(2))
      IF (AZV(2).LT.0.) GO TO 160
      AZV(3) = APT
      NGOOD = 1
      DDIS(1) = .00000001
C        FOR ST. LOUIS USE ID(2)  NOT  ID(1)
      KD = ID(2)
      AZV(4) = XID
      RETURN
150   CONTINUE
      WRITE(LOUT,2) ID,KOLD(2)
      GO TO 170
C
```

```
C DATA RECORDS HAVE GAPS, TRY NEIGHBORING STATION(S)                    MM
160   CONTINUE                                                          MM
      WRITE(LOUT,3) ID,KOLD                                             MM
C                                                                       MM
C ORDER THE STATIONS AND CHECK THEIR DATA                               MM
170   CONTINUE                                                          MM
      CALL WSECLS(XS,YS,DIS,IDIS)                                       MM
      KCT = 0                                                           MM
      NGOOD = 0                                                         MM
180   CONTINUE                                                          MM
      IF (NGOOD.EQ.NC) GO TO 240                                        MM
      KCT = KCT+1                                                       MM
      IF (KCT.GT.NUMSTA) GO TO 240                                      MM
      KUSE = IDIS(KCT)                                                  MM
      NAME(1) = ISTANS(1,KUSE)                                          MM
      NAME(2) = ISTANS(2,KUSE)                                          MM
      DO 190 I=1,NFILES                                                 MM
      IF (KOLD(2).EQ.IDATE(I)) GO TO 200                                MM
190   CONTINUE                                                          MM
      GO TO 120                                                         MM
200   CONTINUE                                                          MM
C                                                                       MM
C         ***********  FOR ST. LOUIS ONLY  ******************************MM
C                                                                       MM
                                                                        MM
C         REMOVE CARDS BETWEEN THESE LINES FOR OTHER CITIES             MM
      I = 1                                                             MM
      GO TO 215                                                         MM
 205  CONTINUE                                                          MM
C                                                                       MM
C         **************************************************************MM
      IC = 0                                                            MM
      IF = IFILE + I                                                    MM
      IF(INCOR.EQ.IDATE(I)) GO TO 215                                   MM
      REWIND IF                                                         MM
210   CONTINUE                                                          MM
      IC = IC + 1                                                       MM
        IF(IC.GT.ICT(I)) GO TO 214                                      MM
      READ (IF) NWDTE,NAM,APT,PT,XIMPH                                  MM
      NWDATA(1,IC) = NWDTE                                              MM
      NWDATA(2,IC) = NAM(1)                                             MM
      NWDATA(3,IC) = NAM(2)                                             MM
      AWDATA(4,IC) = APT                                                MM
      CALL XMIT(24,PT,AWDATA(5,IC))                                     MM
      CALL XMIT(24,XIMPH,AWDATA(29,IC))                                 MM
      GO TO 210                                                         MM
C                                                                       MM
214   INCOR = IDATE(I)                                                  MM
215   CONTINUE                                                          MM
C                                                                       MM
C     SEARCH FOR STATION                                                MM
C                                                                       MM
      M = ICT(I)                                                        MM
      DO 217 K = 1,M                                                    MM
      NAM(1) = NWDATA(2,K)                                              MM
      NAM(2) = NWDATA(3,K)
      IF(NAME(1).EQ.NAM(1).AND.NAME(2).EQ.NAM(2)) GO TO 220
217   CONTINUE
      GO TO 180
220   CONTINUE
      APT = AWDATA(4,K)
      CALL XMIT(24,AWDATA(5,K),PT)
      CALL XMIT(24,AWDATA(29,K),XIMPH)
C
      CALL EXTRP(KOLD,PT,ZAZ)
      IF (ZAZ.LT.0.) GO TO 180
      CALL EXTRP(KOLD,XIMPH,VELL)
      IF (VELL.LT.0.) GO TO 180
      AZV(4*NGOOD+1) = ZAZ
      AZV(4*NGOOD+2) = VELL
      AZV(4*NGOOD+3) = APT
      KD = NAM(2)
      AZV(4*NGOOD+4) = XID
      DDIS(NGOOD+1) = SQRT(DIS(KCT))
      IF (DIS(KCT).LE.RMIN) GO TO 230
      IF (DIS(KCT).GT.RMAX) GO TO 240
C
C PERFORM ANGLE TEST
      CALL ANGTST(XS,YS,NGOOD,KUSE,OK)
      IF (OK.LT.0.) GO TO 180
C
C PERFORM THE BARRIER TEST
      XT = SSTAN(1,KUSE)
      YT = SSTAN(2,KUSE)
      CALL BARIER(XT,YT,XS,YS,OK)
      IF (OK.LT.0.) GO TO 180
      NGOOD = NGOOD+1

      GO TO 180
230   CONTINUE
      NGOOD = 1
240   CONTINUE
C
C     CHECK WHETHER DISTANCES ARE EQUAL TO ZERO
C
      IF(NGOOD.EQ.0) RETURN
      IF(DDIS(1).EQ.0.) DDIS(1) = 1.0E-8
      IF(NGOOD.EQ.1) RETURN
      DO 250 I = 2,NGOOD
      IF(DDIS(I).NE.0.) GO TO 250
      WRITE(LOUT,4) XS,YS
      STOP
250   CONTINUE
      RETURN
1     FORMAT (//,33H GETAZV CANT FIND DATE FILE FOR  ,I6,
     1             22H  IN WIND DATA TABLES.,//)
2     FORMAT (/,54H GETAZV CANT FIND WIND DATA FOR THIS STATION AND DATE
     1. ,2X,A4,A4,2X,I6,/,32H WILL CHECK NEAREST NEIGHBOR(S).,/)
3     FORMAT (/,51H INCOMPLETE DATA PREVENTS USE OF THIS STATION DATA.,
     1  2X,A4,A4,2X,I6,2X,I6,/,30H MUST USE NEAREST NEIGHBOR(S).,/)
```

```
4     FORMAT (1H0,2X,33HMULTIPLE WIND STATIONS FOUND AT (,F8.2,1H,,
     1        F8.2,15H). JOB ABORTED.)
      END



      SUBROUTINE GETCON (KNEW,NC,IRF)
C
C       GETCON FINDS THE 'NC' CLOSEST STATIONS WITH DATA FOR EACH
C       VARIABLE IN THE AIR QUALITY DATA ARRAY (CON) AND
C       DETERMINES THE INTERPOLATED VALUES.
C
      DIMENSION  KNEW(2), IDD(2), ISTUSE(3,10), ACON(4,10),
     1IWORK(25), WORK(25),  R(3),  W(3)   , DIST(3)
      COMMON /AIRQAL/ NEEDAQ,   NOSPEC,   CONAM(10),   CON(24,25,10)
      COMMON /INPUTS/ TITLE(20),        JDATE(10),          NCURV
      COMMON /KZINPT/ NEEDKZ,           KZPRTX,             NUMSD,
     1                TIMESD(3),        TEMPSD(50,3),       ZELVSD(50,3)
     2          ,     TEMPSF(24),       NPTSD(3)       ,    NKZDAT
      COMMON /REUSE/  KPSTAT, KPWDAT, CONVRT, KXTRA, KPWIND
      COMMON /WDATA/  NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      COMMON /WINFLD/ AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)
      EQUIVALENCE (WORK,IWORK)
      INTEGER ATSTAT
      DATA  LOUT /6/
      DATA IYES  /4HYES /
      DATA NONE /4HNONE/
      N = NOSPEC
      IHR = KNEW(1)
      NCP = NC + 1
C
      N1 = N*4
      CALL XMIT(-N1,-1.00,ACON)
      N1 = N1- N
      CALL XMIT(-N1,NONE,ISTUSE)
C
      ATSTAT = NONE
      IF(DISAVE(1).GT. RMIN)  GO TO 90
      ATSTAT = IYES
      KD = IDSAVE(1)
   90 CONTINUE
      DO 300 JS = 1,N
      KSUC = 0

      KTRY = 0
      IF(ATSTAT.NE.IYES)  GO TO 100
C       THE TRAJECTORY IS AT A STATION
      CALL XMIT (24,CON(1,KD,JS),WORK)
      CALL EXTRP (IHR,WORK,VAL)
      IF (VAL.LT.0.)  GO TO 100
C       INTERPOLATION OF DATA FROM OTHER STATIONS IS UNNECESSARY.
      ISTUSE(1,JS) = ISTANS(2,KD)
      ACON(1,JS) =     VAL
      ACON(NCP,JS) =  VAL
      GO TO 300
```

```
C
  100 CONTINUE
C     FIND AND CHECK DATA FROM CLOSEST STATIONS.
  110 KTRY = KTRY + 1
      IF(KTRY.GT.NUMSTA) GO TO 120
      IF(DISAVE(KTRY) .GT. RMAX)  GO TO 130
      KT = IDSAVE(KTRY)
      CALL XMIT(24, CON(1,KT,JS), WORK )
      CALL EXTRP (IHR,WORK,VAL)
      IF(VAL.LT.0.0)  GO TO 110
      KSUC = KSUC + 1
      ISTUSE(KSUC,JS) = ISTANS(2,KT)
      ACON(KSUC,JS) = VAL
      DIST(KSUC) = SQRT( DISAVE(KTRY) )
      IF (KSUC.LT. NC)  GO TO 110
  120 KTRY = MINO(KTRY,NUMSTA)
  130 CONTINUE
      IF(KSUC.LT.1)  GO TO 300
C
C        NOW INTERPOLATE THE VALUES FROM THE CLOSEST STATIONS
C
      SUMR = 0.0
      DO 220  I=1,KSUC
      IF(IRF.EQ. 2)  GO TO 200
      R(I) = 1./DIST(I)
      GO TO 210
  200 R(I) = 1./(DIST(I)**2)
C
  210 SUMR = SUMR + R(I)
  220 CONTINUE
C
      SUMW = 0.0
      DO 230 I = 1,KSUC
      W(I) = R(I)/SUMR
      SUMW = SUMW + ACON(I,JS)*W(I)
  230 CONTINUE
      ACON(NCP,JS) = SUMW
C
  300 CONTINUE
C
      IF(KXTRA.NE.IYES)   GO TO 410
      RS = SQRT(RMAX)
      DO 350 I = 1,NUMSTA
      J = IDSAVE(I)
      IWORK(I) = ISTANS(2,J)
      DISAVE(I) = SQRT( DISAVE(I))
      IF(DISAVE(I).GT.RS)  GO TO 360
  350 CONTINUE
 360  KTRYM = I - 1
C
C        WRITE STATION VALUES AND INTERPOLATED VALUES ON FILE.

C
      WRITE(LOUT,1) KNEW(1),  ( IWORK(I),I=1,KTRYM)
      WRITE(LOUT,2) KNEW(2),  (DISAVE(I),I=1,KTRYM)
      WRITE(LOUT,3)   (K,K=1,NC)
      DO 400  JS = 1,N
      WRITE(LOUT,4) CONAM(JS), (ACON(I,JS),ISTUSE(I,JS),I=1,NC),
     1   ACON(NCP,JS)
  400 CONTINUE
      GO TO 420
  410 CONTINUE
      WRITE(LOUT,5)  KNEW
      WRITE(LOUT,6) (CONAM(J), ACON(NCP,J),J=1,N)
  420 CONTINUE
      N1 = 2*RMAX
      CALL XMIT(-NUMSTA,N1,DISAVE)
      CALL XMIT(-NUMSTA, 0,IDSAVE)
      IHR = KNEW(1)/100 + 1
      TEMPSF(IHR) = ACON(NCP,1)
      RETURN
C
    1 FORMAT(1H ,/,3X,6HTIME =,I6,5X,16HSTATION NUMBERS  , 4X,
     1 5(2X,A4),/,4(40X,2X,A4,2X,A4,2X,A4,2X,A4,2X,A4,/))
    2 FORMAT(1H ,/,3X,6HDATE =,I7,4X,16HDISTANCES (KM)   ,4X,5F6.2,/,
     1    4(40X,5F6.2,/))
    3 FORMAT (1H0,3X,9HVARIABLES ,3(5X,1HX,I1,11H   (STAT)   ),
     1  14HX-INTERPOLATED  )
    4 FORMAT(1H ,4X,A4,4X,3(E9.2,2H (,A4,3H)   ),3X,E11.4 )
    5 FORMAT (1H0,4X,6HTIME =, I5,5X,6HDATE = , I7,//,5X,
     1  31HVARIABLE      INTERPOLATED VALUE   )
    6 FORMAT (1H0,7X,A4,10X,E12.4,//,9(8X,A4,10X,E12.4,//))
      END


      SUBROUTINE JULIAN (IDATE,JULDAT)
C
C************************   ST. LOUIS ONLY   **************************
C
C       THIS SUBROUTINE CONVERTS A 6-INTEGER DATE (YR-MO-DAY) IN TO
C       A JULIAN DATE......JULDAT(1)=DAY.........JULDAT(2)=YEAR
C
      DIMENSION MD(11) ,  JULDAT(2)
      DATA MD /31,28,31,30,31,30,31,31,30,31,30/
C
      IY = IDATE/10000
      IM = (IDATE - IY*10000)/100
      ID = IDATE - IY*10000 - IM*100
C
C       CHECK FOR LEAP YEARS
C
      NL = MOD(IY,4)
      IF(NL.LT. 1)   MD(2) = 29
C
      ISUM = 0
      IM = IM - 1
      DO 10 J = 1,IM
      ISUM = ISUM + MD(J)
   10 CONTINUE
```

```
      JULDAT(1) = ISUM + ID
      JULDAT(2) = 1900 + IY
C
      RETURN
      END



      SUBROUTINE KEKLAY(LMAX,ZMAX,ZHIGH,AKMIN,DH)
C

C-----THIS SUBROUTINE DETERMINES THE VALUES OF KZ IN THE EKMAN LAYER
C     IN THE MANNER SUGGESTED BY O'BRIEN(1970).
C
      COMMON/DIFDAT/DELTAT(30),STAB(30),ZMIX(30),USTAR(30),
     +               UZZ(201),AKZ(201)
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)
      DO 229 J=2,LMAX
      ZHT=Z(J-1)
      ZB=0.10*ZMIX(J-1)
      DZB=ZMIX(J-1)-ZB
      IF(STAB(J-1).GT.0.) GO TO 226
  223 ZFREE=- 5.*STAB(J-1)
      IF(ZB.GT.ZFREE) GO TO 224
      AKB=.35*USTAR(J-1)*ZB*(1.-15.*ZB/STAB(J-1))**.25
      DKZB=.35*USTAR(J-1)*((1.-18.75*ZB/STAB(J-1))/(1.-15.*ZB/STAB(J-1))
     +**.75)
      GO TO 227
  224 ZB=ZFREE
      AKB=.35*USTAR(J-1)*ZB*(1.-15.*ZB/STAB(J-1))**.25
      DKZB=0.7*USTAR(J-1)*(-ZB/STAB(J-1))**.3333
      GO TO 227
  226 AKB=.35*USTAR(J-1)*ZB/(1.+4.7*ZB/STAB(J-1))
      DKZB=.35*USTAR(J-1)/(1.+4.7*ZB/STAB(J-1))**2.
  227 CONTINUE
      AKD=AKB-AKMIN
  228 ZHT=ZHT+DH
      IF(ZHT.GT.Z(J)) GO TO 229
      IF(ZHT.LE.ZB) GO TO 228
      NZHT=IFIX(ZHT/DH)+1
      AKZ(NZHT)=AKMIN
      DELZ=ZHT-ZMIX(J-1)
      IF(DELZ.GT.0.) GO TO 228
      DELZ=DELZ*DELZ
      AKZ(NZHT)=AKMIN+DELZ/(DZB**2.)*(AKD+(ZHT-ZB)*(DKZB+2.*AKD/DZB))
      GO TO 228
  229 CONTINUE
  260 LMAXM1=LMAX-1
      DO 270 J=2,LMAXM1
      IF(STAB(J-1).GT.0.) GO TO 270
      NZ=IFIX(Z(J-1)/DH)+1
      NZZ=IFIX(Z(J)/DH)+1
      AKZ(NZZ)=2.*AKZ(NZZ+1)-AKZ(NZZ+2)
      ZHT=Z(J-1)
      IF(ZHT.GE.ZMAX) GO TO 275
  265 ZHT=ZHT+DH
      IF(ZHT.GT.Z(J)) GO TO 270
      ZHZ=((Z(J)-ZHT)/(Z(J)-Z(J-1)))**.33333
      NZHT=IFIX(ZHT/DH)+1
      ADDKZ=(ZHT-Z(J-1))*(AKZ(NZZ)-AKZ(NZ))/(Z(J)-Z(J-1))+AKZ(NZ)
      AKZ(NZHT)=(AKZ(NZHT)-ADDKZ)*ZHZ+ADDKZ
      IF(AKZ(NZHT).LT.AKMIN) AKZ(NZHT)=AKMIN
      GO TU 265
  270 CONTINUE
  275 IF(LMAX.LT.3) GO TO 290
      DO 280 J=3,LMAX
      NZ=IFIX(Z(J-1)/DH)+1
      IF(AKZ(NZ+1).LE.AKZ(NZ)) GO TO 280
      ZHT=Z(J-1)
      ZHTX=Z(J)
      IF(ZMIX(J-1).LT.Z(J)) ZHTX=ZMIX(J-1)
  276 ZHT=ZHT+DH
      IF(ZHT.GT.Z(J)) GO TO 280

      IF(ZHT.GT.ZMIX(J-1)) GO TO 280
      NZHT=IFIX(ZHT/DH)+1
      ZHZ=((ZHT-Z(J-1))/(ZHTX-Z(J-1)))**.25
      ADDKZ=(ZMIX(J-1)-ZHT)*(AKZ(NZ)-AKMIN)/(ZMIX(J-1)-Z(J-1))+AKMIN
      AKZ(NZHT)=(AKZ(NZHT)-ADDKZ)*ZHZ+ADDKZ
      GO TO 276
  280 CONTINUE
  290 RETURN
      END



      FUNCTION KLASS(A)
C
C     *KLASS* DETERMINES THE ATMOSPHERIC STABILITY CLASS
C             FROM THE FULLE STABILITY PARAMETER (A).
      KLASS = 6
      IF(A.LE.11.25) KLASS = 5
      IF(A.LE. 9.50) KLASS = 4
      IF(A.LE. 8.75) KLASS = 3
      IF(A.LE. 8.00) KLASS = 2
      IF(A.LE. 7.00) KLASS = 1
      RETURN
      END



      SUBROUTINE KZDATA(IERR)
C
C       THIS SUBROUTINE READS (UP TO 3) VERTICAL TEMPERATURE PROFILES.
C       THIS SUBROUTINE READS HOURLY SURFACE TEMPERATURES (DEGREES C)
C       WHEN  RDTEMP EQUALS YES.
C
C       NOTE .. FOR ST. LOUIS RDTEMP EQUALS NO SINCE THE SURFACE
C                TEMPERATURES ARE INTERPOLATED ALONG THE TRAJECTORY.
C
```

```
      COMMON /KZINPT/ NEEDKZ,            KZPRTX,              NUMSO,
     1                TIMESD(3),         TEMPSD(50,3),        ZELVSD(50,3)
     2,               TEMPSF(24),        NPTSD(3),            NKZDAT
      COMMON/CELECT/CELHTS(5,1),NOHTS,NMES
      DATA LIN /3/
      DATA YES, RNEG /4HYES ,4HNO  /
      EQUIVALENCE(KYES,YES)
C
      READ(LIN,1) NKZDAT
      IF(NKZDAT.NE.KYES) GO TO 120
      READ(LIN,1)  KZPRTX
C       READ VERTICAL PROFILES
      DO 50  ISD=1,4
      READ(LIN,2)  A
      IF(A.LT.0.)  GO TO 60
      IF(ISD.GE.4)  GO TO 131
      TIMESD(ISD) = A
      DO 30  J=1,51
      IF(J.GE.51)  GO TO 132
   10 READ(LIN,2)  B, C
      IF(B.LT.0.0)   GO TO 40
C************************NOTE TO USER*********************
C
C     CARDS BETWEEN STARRED LINES SHOULD BE REMOVED
C     FOR CITIES OTHER THAN ST. LOUIS.  THESE CARDS
C     REMOVE FROM THE TEMPERATURE PROFILES THE POINTS
C     HAVING HEIGHTS WHICH ARE MULTIPLES OF 100.0 M.
C     IT IS OFTEN STANDARD PROCEDURE TO RECORD THESE
C     HEIGHTS EVENTHOUGH THEY ARE NOT IMPORTANT TO
C     THE SHAPE OF THE SOUNDING.
      IF(J.EQ.1) GO TO 20
      IB=IFIX(B*1000.+.1)
      IF(MOD(IB,100000).EQ.0) GO TO 10
C**********************************************************
   20 ZELVSD(J,ISD) = B

      TEMPSD(J,ISD) = C
  30  CONTINUE
      GO TO 131
  40  NPTSD(ISD) = J - 1
  50  CONTINUE
      GO TO 132
   60 CONTINUE
      NUMSD=ISD-1
      READ(LIN,1)  RDTEMP
      IF(RDTEMP.NE.YES)  GO TO 90
C       READ SURFACE TEMPERATURES
      DO 80 I= 1,25
      READ(LIN,2) A, B
      IF(A.LT.0.0)  GO TO 90
      IS = IFIX((A+1.E-3)/100.)+1
      IF(IS.GE.25)  GO TO 133
      TEMPSF(IS) = B
   80 CONTINUE
      GO TO 133
```

```
C
   90 CONTINUE
      READ(LIN,4) NOHTS
      READ(LIN,5)    (CELHTS(I,1),I=1,NOHTS)
      NMES = 1
C
  120 IERR=1
      RETURN
C
  131 IERR=-1
      RETURN
  132 IERR=-2
      RETURN
  133 IERR=-3
      RETURN
  134 IERR=-4
      RETURN
C
    1 FORMAT (40X,A4)
    2 FORMAT (40X,2F10.0)
    4 FORMAT(40X,I2)
    5 FORMAT(40X,F10.0)
      END


      SUBROUTINE METIN (IDT,LTAPE)
C
C        THIS IS THE SHORT RECORD  VERSION OF METIN
C
C        THIS SUBROUTINE READS WIND SPEEDS AND DIRECTIONS, SURFACE
C        TEMPERATURES, AIR QUALITY DATA, AND SOLAR RADIATION DATA
C        FROM ST. LOUIS RAMS TAPES. .
C
C        THE WIND DATA IS STORED IN AWDATA.
C        THE TEMPERATURE DATA IS STORED IN CON(I,J,1) AND TEMPSF(I).
C        THE AIR QUALITY DATA IS STORED IN CON(I,J,2 THRU 10).
C        THE RADIATION DATA IS REDUCED INTO UV(I) AND PRINTED ON LOUT.
C
C        NOTE......ONLY ONE DAYS WORTH OF DATA IS STORED AT A TIME.
C        THIS SUBROUTINE  FINDS THE DATA FOR THE DATE (IDT) SPECIFIED.
C
C
      DIMENSION AIMAGE(662) , WORK(550),  X(25,22), UV(24), JULDAT(2),
     1  NWDATA(52,25) , DTEMP(24) , ASHR(24) , LWDSTA(9), CLOUDY(24)
C

      COMMON /AIRQAL/ NEEDAQ,   NOSPEC,   CONAM(10),   CON(24,25,10)
      COMMON /DATES/  IDATE(10),  ICT(10),  NFILES
      COMMON /INPUTS/ TITLE(20),  JDATE(10),  NCURV
      COMMON /KZINPT/ NEEDKZ,            KZPRTX,            NUMSD,
     1                TIMESD(3),         TEMPSD(50,3),      ZELVSD(50,3)
     2        ,       TEMPSF(24),        NPTSD(3)      ,    NKZDAT
      COMMON /REUSE/  KPSTAT, KPWDAT, CONVRT, KXTRA, KPWIND
      COMMON /WDATA/  NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      COMMON /WHERE/  RLAT,   RLONG,     TMZONE
      COMMON /WINFLO/ AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)
C
      EQUIVALENCE (YES,IYES)
      EQUIVALENCE (X,WORK)
      EQUIVALENCE (NWDATA,AWDATA)
      DATA IYES, NO /4HYES ,4HNO  /
      DATA ZERO /0.0000000/
      DATA LIN,LOUT,LPUNCH /3,6,1/
C
C         ***********DATA SPECIFICALLY FOR ST. LOUIS**********************
      DATA NODAYS /1/
      DATA NRAM /4HRAMS/
      DATA APT / 90./
      DATA HIVAL / 500. /
      DATA LWDSTA  / 108,110,114,115,116,117,118,121,0/
      DATA NSTMAX , NSPMAX /25, 10/
      DATA CONAM /4HTEMP, 4HO3  , 4HCO  , 4HCH4 , 4HTHC , 4HNO  , 4HNO2
     1  ,     4HNOX , 4HTS  , 4HSO2  /
 6    NOSPEC = 10
      DEBUG = NO
C         ****************************************************************
      LU = LTAPE
      KNTER = 0
      NFILE = 0
      IMIN = 1
      IMAX = 24
C
      IDATE(1) = IDT
      ICT(1) = NUMSTA
      NFILES = 1
C
C
      CALL JULIAN(IDT,JULDAT)
C
C        INITIALIZE ARRAYS
C
      IF(IMIN.EQ.1.AND.IMAX.EQ.24)  GO TO 95
C
      CALL XMIT (24,-1.,UV)
      N = 52*NSTMAX
      CALL XMIT(-N,-1.,AWDATA)
      N = 24*NSPMAX*NSTMAX
      CALL XMIT(-N,-1.,CON)
C
  95  CONTINUE
C
      DO 100 J = 1,NUMSTA
      NWDATA(1,J) = IDT
      NWDATA(2,J) = ISTANS(1,J)
      NWDATA(3,J) = ISTANS(2,J)
      AWDATA(4,J) = APT
 100  CONTINUE
C
```

```
C         RETRIEVE ONE HOURS WORTH OF DATA FROM TAPE
C
  110 CONTINUE
C
C          ON UNIVAC ELIMINATE CARDS BETWEEN STARED CARDS
C          ELIMINATE AIMAGE FROM DIMENSION STATEMENT ABOVE
C          **************************************************************
      BUFFER IN (LU,0) (AIMAGE(1),AIMAGE(2))
  120 IF(UNIT,LU)  120, 130, 460, 470
C            NOT READY  ....READY...EOF...PARITY
  130 CONTINUE
      DECODE(12,20,AIMAGE(1))  IYR,  IDAY,  IHR
C          **************************************************************
C          ON UNIVAC INCLUDE THE FOLLOWING CARD
C     READ (LU,20,ERR=470,END=460)  IYR, IDAY, IHR
      IF(IDAY.EQ.JULDAT(1))  GO TO 140
C      SKIP FILE LU
      NFILE = NFILE + 1
      IF(NFILE.GE.NODAYS)  GO TO 450
      GO TO 110
  140 CONTINUE
C          ON UNIVAC ELIMINATE CARDS BETWEEN STARED CARDS
C          **************************************************************
      I1 = 3
      I2 = I1 + 29
      DO 145 I = 1,22
      BUFFER IN (LU,0)  (AIMAGE(I1),AIMAGE(I2))
  143 IF(UNIT,LU)  143,  144,  460,  470
  144 I1 = I2 + 1
      I2 = I1 + 29
  145 CONTINUE
C          **************************************************************
C          ON UNIVAC INCLUDE THE FOLLOWING CARDS
C     DO 150 K = 1,22
C     READ (LU,40,ERR=470,END=460)  (X(J,K),J=1,NUMSTA)
C150  CONTINUE
      I1 = IMIN - 1
      IF(IHR.LT.I1)  GO TO 110
C          ON UNIVAC ELIMINATE CARDS BETWEEN STARED CARDS
C          **************************************************************
      I1 = 3
      J1 = 1
      J2 = 10
      DO 150 K = 1,55
      DECODE(120,40,AIMAGE(I1))  (WORK(JS),JS=J1,J2)
      J1 = J1 + 10
      J2 = J2 + 10
      I1 = I1 + 12
  150 CONTINUE
C          **************************************************************
      KNTER = KNTER + 1
C
      IF(DEHUG.NE. YES)  GO TO 170
      WRITE(LOUT,30) IYR, IDAY, IHR
      DO 160 K = 1,22
      WRITE(LOUT,50)  K, (X(J,K),J=1,NUMSTA)
 160  CONTINUE
 170  CONTINUE
C
 300  INDEX = IHR + 1
C
C

C        XMIT WIND DATA
C
      I1 = INDEX + 4
      I2 = INDEX + 28
      DO 310 J = 1,NUMSTA
      AWDATA(I1,J) = X(J,2)
      AWDATA(I2,J) = X(J,1)
 310  CONTINUE
C        XMIT TEMPERATURE DATA
      DO 320 J = 1,NUMSTA
      CON(INDEX,J,1) = X(J,3)
 320  CONTINUE
      IF(NEEDAQ.NE.IYES)   GO TO 350
C
C        REDUCE SOLAR RADIATION DATA
C        CURRENTLY ALL SOLAR RADIATION DATA BELOW .05 LANGLEYS/MIN
C        AND ABOVE 2.0 LANGLEYS/MIN IS EXCLUDED.
C
      UV(INDEX) = 0.0
      SUM1 = 0.0
      SUM2 = 0.0
      KS = 0
      DO 330 J= 1,NUMSTA
      IF( X(J,7).LT. 0.05)  GO TO 330
      IF( X(J,7).GT. 2.00)  GO TO 330
      IF( X(J,8).LT. 0.05)  GO TO 330
      IF( X(J,8).GT. 2.00)  GO TO 330
      SUM1 = SUM1 + X(J,7)
      SUM2 = SUM2 + X(J,8)
      KS = KS + 1
 330  CONTINUE
      IF(KS.GT.0)  UV(INDEX) = (SUM1 - SUM2)/KS
C
C        XMIT TEMPERATURE AND ATMOSPHERIC CONCENTRATION DATA
C
      DO 340  I = 2,10
      K = I + 9
      IF(I.GE.10)  K = I + 10
      DO 340 J = 1,NUMSTA
      CON(INDEX,J,I) = X(J,K)
 340  CONTINUE
 350  CONTINUE
C
C         DETERMINE HOURLY AVERAGE NEAR-SURFACE TEMPERATURE GRADIENTS
C         AND WIND SHEAR CORRECTION FOR 10M ELEVATION WIND SPEEDS.
C
C         NEUTRAL CONDITIONS   (DEFAULT CONDITION)
```

```
      ASHW(INDEX) = 1.219
      SUM1 = 0.0
      KS = 0
      DO 360 J = 1,NUMSTA
      IF(X(J,6) .GT. 10.  )   GO TO  360
      IF(X(J,6) .LT.-10.  )   GO TO 360
      SUM1 = SUM1 + X(J,6)
      KS = KS + 1
  360 CONTINUE
      IF(KS.LT.1)  GO TO 370
      DTEMP(INDEX) = SUM1/25./KS
C           UNSTABLE CONDITIONS
      IF(DTEMP(INDEX).LT. -.015)  ASHR(INDEX) = 1.166
C           STABLE CONDITIONS
      IF(DTEMP(INDEX) .GT. -.005)  ASHR(INDEX) = 1.437

C
 370  CONTINUE
C
      IF(KNTER.LT. IMAX)   GO TO 110
C
C          YOU ARE FINISHED READING THE DATA FROM THE TAPE.
C
C          EDIT WIND DATA AND CONVERT THE WIND DIRECTIONS FROM THE
C          RAMS 360 POINT SYSTEM TO A 90 POINT SYSTEM
C
 380  CONTINUE
      KS = 1
      A = APT/360.0
      AINV = 1./A
      DO 390 J = 1,NUMSTA
      K = LWDSTA(KS) - 100
      DO 390 I = IMIN , IMAX
      I1 = I + 4
      I2 = I + 28
      IF(AWDATA(I1,J).GT.HIVAL) AWDATA(I1,J) = - AINV
      AWDATA(I1,J) = AWDATA(I1,J)*A
      IF(AWDATA(I2,J).GT.HIVAL)    AWDATA(I2,J) = -1.0
C           AND ADJUST 10M WIND SPEEDS TO THE 30M STANDARD
      IF(J.NE.K)    GO TO 390
      AWDATA(I2,J) = AWDATA(I2,J)*ASHR(I)
      KS = KS + 1
 390  CONTINUE
      IF(NEEDAQ.NE.IYES)    GO TO  440
C
C          EDIT ATMOSHPERIC CONCENTRATION DATA
C
      DO 410 K = 2, NOSPEC
      DO 410 J = 1, NUMSTA
      DO 400 I = IMIN, IMAX
      IF(CON(I,J,K).GT.HIVAL)  CON(I,J,K) = -1.0
 400  CONTINUE
 410  CONTINUE
C
C          EDIT AND REDUCE TEMPERATURE DATA
```

```
C
      DO 430 I = IMIN, IMAX
      KS = 0
      SUM1 = 0.
      DO 420 J = 1, NUMSTA
      IF(CON(I,J,1).GT.HIVAL)   CON(I,J,1) = -1.0
      IF(CON(I,J,1).LT.ZERO)  GO TO 420
      SUM1 = SUM1 + CON(I,J,1)
      KS = KS + 1
 420  CONTINUE
      IF(KS.GT.0)  TEMPSF(I) = SUM1/KS
 430  CONTINUE
 440  CONTINUE
C
C          GENERATE SKY CLEARNESS RATIOS FROM UV DATA
C
      CALL SKY(UV,CLOUDY,IDATE)
C
      REWIND LTAPE
C
      GO TO 480
 450  WRITE(LOUT,60)  NFILE,IYR,IDAY,IHR
      GO TO 500

 460  WRITE(LOUT,70)  IYR,  IDAY,  IHR
      GO TO 500
 470  WRITE(LOUT,80)     IYR,  IDAY,  IHR
      GO TO 500
C
 480  CONTINUE
C        PRINT WIND DATA IF KPWDAT = YES
C        PUNCH WIND DATA IF SPCHWD = YES
      SPCHWD = NO
      IF (KPWDAT.NE.IYES)  GO TO 500
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,77)
      DO 490 J = 1,NUMSTA
      WRITE(LOUT,78) (NWDATA(I),I=1,28)
      WRITE(LOUT,79) (NWDATA(I),I=1,3) , (NWDATA(I),I=29,52)
      IF(SPCHWD.NE.IYES)  GO TO 490
      WRITE(LPUNCH,73) (NWDATA(I),I=1,28)
      WRITE(LPUNCH,74) (NWDATA(I),I=1,3), (NWDATA(I),I=29,52)
 490  CONTINUE
 500  CONTINUE
C
      IF(DEBUG.NE.YES)  GO TO 530
C
      WRITE(LOUT,82)
      DO 510 J = 1,NUMSTA
      WRITE(LOUT,85) J, (NWDATA(I,J),I=1,3), AWDATA(4,J)
      WRITE(LOUT,90)    (AWDATA(I,J), I=5,52)
 510  CONTINUE
C
      WRITE(LOUT,84)
      DO 520 K = 1,NOSPEC
      WRITE(LOUT,86)   CONAM(K),  CONAM(K),  CONAM(K), CONAM(K)
      DO 520 J = 1,NUMSTA
      WRITE(LOUT,85) J, (NWDATA(I,J),I=1,3)
      WRITE(LOUT,90) (CON(I,J,K), I=1,24)
  520 CONTINUE
C
  530 CONTINUE
      ICASE = 1
      RETURN
C
      ENTRY EMETIN
      IF(NEEDAQ.NE.IYES) RETURN
      IF(ICASE.GT.1)  GO TO 555
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,89)  IYR,IDAY
      WRITE(LOUT,87)
      DO 540 I=1,24
      I1 = (I-1)*100
      I2 = I1 + 59
      WRITE(LOUT,88) I1, I2, UV(I), TEMPSF(I), DTEMP(I), CLOUDY(I)
  540 CONTINUE
C
C         PUNCH CLEARNESS RATIOS AND TEMPERATURES FOR INPUT TO KEMOD
      IF(KPWIND.NE.IYES) RETURN
      A = -10.0
      WRITE(LPUNCH,81)  TITLE
      DO 550 I = 6,20
      TM = 60.*FLOAT(I-1)
      WRITE(LPUNCH,75)  TM, CLOUDY(I)
 550  CONTINUE
      WRITE(LPUNCH,75)  A

 555  CONTINUE
      DO 560 I =6,20
      TM = 60.*FLOAT(I-1)
      WRITE(LPUNCH,76)  TM, TEMPSF(I)
 560  CONTINUE
      WRITE(LPUNCH,76)  A
      ICASE = ICASE + 1
 580  RETURN
C
   20 FORMAT (3I4)
   30 FORMAT (10X,4I10)
   40 FORMAT (10E12.6)
   50 FORMAT (1H0,2HV=,I3,4X,10E11.3,/,10X,10E11.3,/,10X, 5E11.3)
   60 FORMAT (1H0, 38HTHE DATE REQUESTED IS NOT IN THE FILE       ,4I8)
   70 FORMAT (1H0, 38HAN UNEXPECTED EOF  --  METIN STOPPED        ,4I8)
   73 FORMAT (I6,1X,2A4,13X,I2,2HPT,24I2)
   74 FORMAT (I6,1X,2A4,14X,3HM/S/,24I2)
   75 FORMAT (20H CLEARNESS RATIOS        ,20X,2F10.3)
   76 FORMAT (20H TEMPERATURE (DEG C)     ,20X,2F10.3)
   77 FORMAT (25H0 EDITED WIND DATA                    )
   78 FORMAT (1H ,I6,1X,2A4,13X,I2,2HPT,24I3)
   79 FORMAT (1H ,I6,1X,2A4,14X,3HM/S,24I3)
   80 FORMAT (1H0, 38HPARITY ERROR ENCOUNTERED  -- METIN          ,4I8)
   81 FORMAT (20A4)                                                     MM  2188
   85 FORMAT (1H0,I2,10X,I6,4X,A4,I4,F10.1 )                            MM  2189
   82 FORMAT (1H1,22HWIND DATA FROM AWDATA    )                         MM  2190
   84 FORMAT(1H1,47H TEMPERATURE AND ATMOSHPERIC CONCENTRATIONS       ) MM  2191
   86 FORMAT(1H0,4(A4,20X)  )                                           MM  2192
   87 FORMAT (1H0,10X,26HREGIONAL HOURLY AVERAGES      ,// 11X,4HHOUR,  MM  2193
     1  10X,22HULTRAVIOLET RADIATION  ,4X,11HTEMPERATURE  ,4X,          MM  2194
     2  20HTEMPERATURE GRADIENT  ,4X,15HCLEARNESS RATIO  ,              MM  2195
     3  /,25X,22H   (LANGLEYS/MIN)        ,                             MM  2196
     5  4X, 11H   (DEG C)      , 9X,14H(DEG C/METER)            )       MM  2197
   88 FORMAT (1H0,8X,I4,3H - ,I4,5X,E15.3,F20.2,F20.4,F20.3 )           MM  2198
   89 FORMAT(1H0,10X,21HST. LOUIS      YEAR =  ,I5,  5X,6HDAY =   ,I3 ) MM  2199
   90 FORMAT (1H ,5X,12E10.2)                                           MM  2200
C                                                                       MM  2201
      END                                                               MM  2202


      SUBROUTINE PLACIT(DT,L,VX,VY,POS,NDX)                             MM  2203
C                                                                       MM  2204
C         STARTING FROM THE POSITION DEFINED BY LOCATION  L  IN THE  P  MM  2205
C         ARRAY,  PLACIT  COMPUTES THE POSITION  POS  AND GRID SQUARE   MM  2206
C         INDICES  NDX  AT TIME  T(L) + DT.  VX  AND  VY  SHOULD BE     MM  2207
C         THE  X  AND  Y  COMPONENTS OF  V(L).                          MM  2208
C                                                                       MM  2209
      DIMENSION POS(2),NDX(2)                                           MM  2210
      COMMON /WIND/ T(100), V(100), TH(100), NPTS                       MM  2211
      COMMON /TRAJ/ TSTART, P(2,100), ID(2,100)                         MM  2212
      COMMON /GRID/ X1,X2,Y1,Y2,NX,NY,DELX,DELY,DELT                    MM  2213
      DATA SMALL /1.E-4/                                                MM  2214
      POS(1) = P(1,L)+VX*DT                                             MM  2215
      POS(2) = P(2,L)+VY*DT                                             MM  2216
      IF(V(L).EQ.0.) GO TO 10                                           MM  2217
      YPLUS=0.0
      YPLUS=0.0
      GO to 20
   10 XPLUS=(VX/V(L))*SMALL
      YPLUS=(VY/V(L))*SMALL
   20 X=(POS(1)+ XPLUS)IDELX
      IADD=1
      IF(X.LT.0.)IADD=0
      NDX(1)=IFIX(X)+IADD
      X=(POS(2)+YPLUS)/DELY
      IADD=1
      IF(X.LT.0.)IADD=0
      NDX(2)=IFIX(X)+IADD
      RETURN
      END


      SUBROUTINE RUFNES(Z0,J)                                           MM  2225
C                                                                       MM  2226
                                                                        MM  2227
C-----THIS SUBROUTINE CHECKS IF THE LOCATION ALONG THE TRAJECTORY       MM  2228
C     AT EACH HOUR IS WITHIN A PRESPECIFIED DISTANCE FROM THE           MM  2229
C     CENTER OF THE URBAN AREA OF THE REGION.  IF SO, A SPECIAL         MM  2230
C     URBAN ROUGHNESS LENGTH (Z0) IS USED RATHER THAN A TYPICAL         MM  2231
C     RURAL ROUGHNESS LENGTH.                                           MM  2232
C                                                                       MM  2233
C     NOTE..........THE UTM X AND Y COORDINATES OF THE CENTER OF        MM  2234
C     THE URBAN AREA (IN KM) AND AN APPROXIMATE RADIAL DISTANCE         MM  2235
C     FROM THIS CENTER MUST BE INPUT IN THE DATA STATEMENT BELOW.       MM  2236
```

```
C
      COMMON/PEVE/PE(2,24),VE(24)
      COMMON/ORIGIN/UTMXOR,UTMYOR
C
      DATA UTMXUR,  UTMYUR  / 739.5, 4280.5/
      DATA RADURB, RADSUB  /5.,10./
      DATA ZOURBN, ZORURL  /1.,.25/
C
      PE(1,J)=PE(1,J)+UTMXOR
      PE(2,J)=PE(2,J)+UTMYOR
      A=(PE(1,J)-UTMXUR)**2
      B=(PE(2,J)-UTMYUR)**2
      A = SQRT(A+B)
      IF(A.GE.RADSUB)  GO TO 40
      IF(A.LE.RADURB)  GO TO 30
      PORT = (A - RADURB)/(RADSUB - RADURB)
      Z0 = ZOURBN + PORT*(ZORURL-ZOURBN)
      GO TO 50
   30 Z0 = ZOURBN
      GO TO 50
   40 Z0 = ZORURL
   50 RETURN
      END


      SUBROUTINE SETIN
C
C       SETIN FOR ST. LOUIS
C
C     SETIN INITIALIZES WIND STATION NAMES AND COORDINATES.
C     SETIN INITIALIZES LATITUDE, LONGITUDE, AND TIME ZONE.
C
C     ISTANS IS AN ARRAY CONTAINING THE NAMES OF THE WIND STATIONS
C       SSTAN IS AN ARRAY CONTAINING THE UTM X-Y COORDINATES OF THE
C       METEROLOGICAL STATIONS
C
C     NUMSTA IS THE NUMBER OF STATIONS
C     RLAT IS THE LATITUDE
C     RLONG IS THE LONGITUDE
C     TMZONE IS THE TIME ZONE (IE
C                                       6.   FOR ST. LOUIS
C                                       8.   FOR LOS ANGELES )
C
      COMMON /WDATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      COMMON  /GRID/  X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT
      COMMON  /ORIGIN/  UTMXOR, UTMYOR
      COMMON /WHERE/  RLAT,    RLONG,     TMZONE
C
C         -------------------------------------------------------
C
C             USER MUST DEFINE THE FOLLOWING INFORMATION
C
      DATA RLAT /38.6/
      DATA RLONG /90.2/                                                 MM 2289
                                                                        MM 2290
      DATA TMZONE /6.0/                                                 MM 2291
      DATA NUMSTA /25/                                                  MM 2292
C                                                                       MM 2293
      DATA ISTANS  /4HRAMS,   4H 101,                                   MM 2294
     1              4HRAMS,   4H 102,   4HRAMS,   4H 103,               MM 2295
     2              4HRAMS,   4H 104,   4HRAMS,   4H 105,               MM 2296
     3              4HRAMS,   4H 106,   4HRAMS,   4H 107,               MM 2297
     4              4HRAMS,   4H 108,   4HRAMS,   4H 109,               MM 2298
     5              4HRAMS,   4H 110,   4HRAMS,   4H 111,               MM 2299
     6              4HRAMS,   4H 112,   4HRAMS,   4H 113,               MM 2300
     7              4HRAMS,   4H 114,   4HRAMS,   4H 115,               MM 2301
     8              4HRAMS,   4H 116,   4HRAMS,   4H 117,               MM 2302
     9              4HRAMS,   4H 118,   4HRAMS,   4H 119,               MM 2303
     *              4HRAMS,   4H 120,   4HRAMS,   4H 121,               MM 2304
     1              4HRAMS,   4H 122,   4HRAMS,   4H 123,               MM 2305
     2              4HRAMS,   4H 124,   4HRAMS,   4H 125  /             MM 2306
C                                                                       MM 2307
C             STAT      UTM-X      UTM-Y                                MM 2308
C                                                                       MM 2309
       SSTAN(1, 1) =  744.183                                           MM 2310
       SSTAN(2, 1) =            4279.862                                MM 2311
       SSTAN(1, 2) =  742.518                                           MM 2312
       SSTAN(2, 2) =            4286.045                                MM 2313
       SSTAN(1, 3) =  747.588                                           MM 2314
       SSTAN(2, 3) =            4282.467                                MM 2315
       SSTAN(1, 4) =  747.312                                           MM 2316
       SSTAN(2, 4) =            4277.304                                MM 2317
       SSTAN(1, 5) =  743.706                                           MM 2318
       SSTAN(2, 5) =            4276.453                                MM 2319
       SSTAN(1, 6) =  738.660                                           MM 2320
       SSTAN(2, 6) =            4277.566                                MM 2321
       SSTAN(1, 7) =  740.179                                           MM 2322
       SSTAN(2, 7) =            4282.610                                MM 2323
       SSTAN(1, 8) =  748.407                                           MM 2324
       SSTAN(2, 8) =            4291.102                                MM 2325
       SSTAN(1, 9) =  755.802                                           MM 2326
       SSTAN(2, 9) =            4279.886                                MM 2327
       SSTAN(1,10) =  747.209                                           MM 2328
       SSTAN(2,10) =            4272.826                                MM 2329
       SSTAN(1,11) =  738.812                                           MM 2330
       SSTAN(2,11) =            4272.479                                MM 2331
       SSTAN(1,12) =  733.938                                           MM 2332
       SSTAN(2,12) =            4280.913                                MM 2333
       SSTAN(1,13) =  737.738                                           MM 2334
       SSTAN(2,13) =            4289.820                                MM 2335
       SSTAN(1,14) =  744.320                                           MM 2336
       SSTAN(2,14) =            4297.456                                MM 2337
       SSTAN(1,15) =  757.111                                           MM 2338
       SSTAN(2,15) =            4297.799                                MM 2339
       SSTAN(1,16) =  762.777                                           MM 2340
       SSTAN(2,16) =            4290.083                                MM 2341
       SSTAN(1,17) =  760.560                                           MM 2342
       SSTAN(2,17) =            4272.818                                MM 2343
      SSTAN(1,18) =  743,065
      SSTAN(2,18) =              4263,256
      SSTAN(1,19) =  729,759
      SSTAN(2,19) =              4270,547
      SSTAN(1,20) =  723,079
      SSTAN(2,20) =              4285,909
      SSTAN(1,21) =  732,414
      SSTAN(2,21) =              4302,376
      SSTAN(1,22) =  741,631

      SSTAN(2,22) =              4329,223
      SSTAN(1,23) =  777,320
      SSTAN(2,23) =              4286,378
      SSTAN(1,24) =  749,275
      SSTAN(2,24) =              4236,537
      SSTAN(1,25) =  697,445
      SSTAN(2,25) =              4282,240

C         ------------------------------------------------------------
C
C     DO 40 I = 1,NUMSTA
      SSTAN(1,I) = SSTAN(1,I) - UTMXOR
      SSTAN(2,I) = SSTAN(2,I) - UTMYOR

C     CONTINUE
40    X1 = 0.0
      X2 = NX*DELX
      Y1 = 0.0
      Y2 = NY*DELY
      RETURN
      END



      SUBROUTINE SETTUP

C         INITIALIZATION SUBROUTINE FOR GENERATION OF
C         TRAJECTORY GRID SQUARE HISTORIES.
C
C     DIMENSION          UTM(2,5)
      COMMON /CNTROL/  KSTOP,  TSUN
      COMMON /GRID/    X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT
      COMMON /INPUTS/  TITLE(20),   JDATE(10),    NCURV
      COMMON /TRAJ/    TSTART,  P(2,100),   ID(2,100)
      COMMON /WDATA/   NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      COMMON /WIND/    T(100),  V(100),  TH(100),  NPTS
      COMMON /ORIGIN/  UTMXOR,    UTMYOR
      DATA LIN, LOUT /3,6/

C         CONVERT WIND VELOCITIES TO KM/MIN AND TIMES TO MINUTES
C
C     V(1) = V(1)/60.
      TSTART = T(1) + TSUN
      TSHIFT = TIMIN(T(1))
      T(1) = 0.000
      DO 100 K = 2,NPTS                                                 MM  2396
      V(K) = V(K)/60.0                                                  MM  2397
      T(K) = TIMIN(T(K)) - TSHIFT                                       MM  2398
      IF(T(K).LT.T(K-1))  T(K) = T(K) + 1440.0                          MM  2399
  100 CONTINUE                                                          MM  2400
C                                                                       MM  2401
C          SET UP GRID AND LOCATE TRAJECTORY START POINT                MM  2402
      X1 = 0.0                                                          MM  2403
      X2 = NX*DELX                                                      MM  2404
      Y1 = 0.0                                                          MM  2405
      Y2 = NY*DELY                                                      MM  2406
C          NORTH-WEST CORNER                                            MM  2407
      UTM(1,1) = UTMXOR                                                 MM  2408
      UTM(2,1) = UTMYOR + Y2                                            MM  2409
C          NORTH-EAST CORNER                                            MM  2410
      UTM(1,2) = UTMXOR + X2                                            MM  2411
      UTM(2,2) = UTMYOR + Y2                                            MM  2412
C          SOUTH-EAST CORNER                                            MM  2413
      UTM(1,3) = UTMXOR + X2                                            MM  2414
      UTM(2,3) = UTMYOR                                                 MM  2415
                                                                        MM  2416
C          SOUTH-WEST CORNER                                            MM  2417
      UTM(1,4) = UTMXOR                                                 MM  2418
      UTM(2,4) = UTMYOR                                                 MM  2419
C          START POINT                                                  MM  2420
      UTM(1,5) = UTMXOR + P(1,1)                                        MM  2421
      UTM(2,5) = UTMYOR + P(2,1)                                        MM  2422
      CALL PLACIT (0.,1,0.,0.,UTM(1,5),JDATE(9))
      ID(1,1) = JDATE(9)
      ID(2,1) = JDATE(10)

                                                                        MM  2425
C          PRINT GRID CORNER AND START POINT COORDINATES                MM  2426
C                                                                       MM  2427
      WRITE(LOUT,3)                                                     MM  2428
      WRITE(LOUT,4)       UTM(1,1), UTM(2,1), X1, Y2                    MM  2429
      WRITE(LOUT,5)       UTM(1,2), UTM(2,2), X2, Y2                    MM  2430
      WRITE(LOUT,6)       UTM(1,3), UTM(2,3), X2, Y1                    MM  2431
      WRITE(LOUT,7)       UTM(1,4), UTM(2,4), X1, Y1                    MM  2432
      WRITE(LOUT,8)       UTM(1,5), UTM(2,5), P(1,1), P(2,1)            MM  2433
C                                                                       MM  2434
C          INITIALIZE PRINTER-PLOT                                      MM  2435
      READ(LIN,1) XL ,XR, YB, YT                                        MM  2436
      CALL SETPLT(XL,XR,YB,YT)                                          MM  2437
      RETURN                                                            MM  2438
C                                                                       MM  2439
C          FORMATS                                                      MM  2440
1     FORMAT(30X,4F10.0)                                                MM  2441
3     FORMAT  (1H0,/////,21H  GRID REFERENCE DATA  ,//, 5X,             MM  2442
     1  12H(KILOMETERS)  ,15X,15HUTM COORDINATES ,8X,17HLOCAL COORDINATEMM  2443
     2S  ,/,34X,1HX,9X,1HY,13X,1HX,9X,1HY )                             MM  2444
4     FORMAT (1H0,4X,17HNORTH WEST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2445
5     FORMAT (1H0,4X,17HNORTH EAST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2446
6     FORMAT (1H0,4X,17HSOUTH EAST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2447
7     FORMAT (1H0,4X,17HSOUTH WEST CORNER  ,6X,2F10.2,4X,2F10.2 )       MM  2448
8     FORMAT (1H0,4X,17HSTART POINT        ,6X,2F10.2,4X,2F10.2 )       MM  2449
      END                                                               MM  2450
      SUBROUTINE SKY (UV,CLOUDY,IDATE)
C
C     GENERATE SKY CLEARNESS RATIOS
C     ST. LOUIS VERSION   12.22.77
C     CONVRT IS SET TO CONVERT WATTS/METER**2 TO LANGLEYS/MIN.
C
      DIMENSION          UV(1),  CLOUDY(1),  IDATE(1)
      COMMON /WHERE/   RLAT,  RLONG,  TMZONE
C
C
      IHALF = 30
      CONVT = 1./697.333334
      IY = IDATE(1)/10000
      IM = IDATE(1)/100 - IY*100
      ID = IDATE(1) - IY*10000  - IM*100
      IY = IY + 1900
      DO 10 I =1,24
      IF(UV(I).GT. 1.E-4)  GO TO 15
      CLOUDY(I) = 1.0
   10 CONTINUE
      GO TO 50
   15 CONTINUE
      IHR = (I-1)*100 + IHALF
      TIME = FLOAT(IHR)
      DO 40 J = I,24
      IF(UV(J).LT. 1.E-4)  GO TO 30
      CALL SOLAR(RLAT,RLONG,TMZONE,IY,IM,ID,TIME,D,5)
      Z = 90.0 - D

      IF(Z.GT.30.)  GO TO 20
      CLEAR = (-.0146667*Z*Z + .05833334*Z + 66)*CONVT
      GO TO 25
   20 CLEAR = (-.992727*Z + 84.4)*CONVT
   25 CLOUDY(J) = UV(J)/CLEAR
      IF(CLOUDY(J).GT. 1.)  CLOUDY(J) = 1.0000
      TIME = TIME + 100
      GO TO 40
   30 CONTINUE
      CLOUDY(J) = 1.000
   40 CONTINUE
   50 RETURN
      END


      SUBROUTINE SMOOTH(LMAX)
C
C-----THIS SUBROUTINE SMOOTHS THE TEMPERATURE SOUNDING BY DIVIDING THE
C     SOUNDING INTO LAYERS WITH LAPSE RATES WHICH ARE GT 0.98, LT 0.98
C     BUT GT 0.0, AND LE 0.0, AND THEN DETERMINING THE AVERAGE LAPSE
C     RATE OVER EACH LAYER.
C
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)
      EQUIVALENCE(JMAXM1,LMAXP1,LMAXM2)
      DIMENSION DELTAT(49)
      LMAX=LMAX-1
      JMAX=LMAX
      M=1
      DO 1050 L=1,LMAX
      DELTAT(L)=(TT(L)-T(L))*100./(ZZ(L)-Z(L))+.98
      IF (L.EQ.1) GO TO 1050
      IF(DELTAT(L))      1025,1025,1030
 1020 M=M+1
      GO TO 1035
 1025 IF(DELTAT(L-1).GT.0.)   GO TO 1020
      GO TO 1035
 1030 IF(DELTAT(L-1).LE.0.) GO TO 1020
      IF((DELTAT(L).GE..98.AND.DELTAT(L-1).LT..98).OR.
     +(DELTAT(L).LT..98.AND.DELTAT(L-1).GE..98)) GO TO 1020
 1035 TT(M)=TT(L)
      ZZ(M)=ZZ(L)
 1050 CONTINUE
      LMAX=M+1
      NO=JMAX-M
      CALL XMIT(-NO,0.,TT(LMAX))
      CALL XMIT(-NO,0.,ZZ(LMAX))
      JMAXM1=JMAX-1
      CALL XMIT(JMAXM1,TT(1),T(2))
      CALL XMIT(JMAXM1,ZZ(1),Z(2))
      JMAX=LMAX
      M=3
      LMAXM2=LMAX-2
      IF(LMAXM2.LT.3)       RETURN
      DO 1100 L=3,LMAXM2
      DELTAT(L)=(TT(L)-TT(L-1))*100./(ZZ(L)-ZZ(L-1))+.98
      IF(DELTAT(L).LT.0..AND.(ZZ(L)-ZZ(L-1)).LT.100.) GO TO 1080
      M=M+1
      GO TO 1100
 1080 NO=LMAX-1-L
      CALL XMIT(NO,TT(L+1),T(M+1))
      CALL XMIT(NO,ZZ(L+1),Z(M+1))
 1100 CONTINUE
      LMAX=M+1
      NO=JMAX-LMAX

      LMAXP1=LMAX+1
      CALL XMIT(-NO,0.,T(LMAXP1))
      CALL XMIT(-NO,0.,T(LMAXP1))
      JMAXM1=JMAX-1
      CALL XMIT(JMAXM1,T(2),TT(1))
      CALL XMIT(JMAXM1,Z(2),ZZ(1))
      RETURN
      END
      SUBROUTINE STABIL(STAB,J,A,ZTOP,DUDZ,DELT,U,ZHTMIX,USTAR,ZW,Z0)
C-----THIS SUBROUTINE CALCULATES THE VALUE OF THE STABILITY PARAMETER A
C     DEVELOPED BY FULLE(1975), AND THEN DETERMINES THE CORRESPONDING
C     VALUE OF THE MONIN-OBUKHOV LENGTH, L.  FIRST, AN APPROPRIATE
C     STABILITY CLASS IS SELECTED ACCORDING TO THE VALUE OF A, AND THEN
C     A VALUE OF L IS DETERMINED AS SUGGESTED BY GOLDER(1972).
C     IN ADDITION, THE HEIGHT OF THE BOUNDARY LAYER IS DETERMINED FOR
C     STABLE, NEUTRAL, AND UNSTABLE CONDITIONS AS A FUNCTION OF U*/F.
C
      COMMON/CORIOL/F
      REAL LASTAB
      DUMB=DUDZ
      U60=U*60.
      IF(J.EQ.1) DUDZ=U/(ZW-Z0)
  100 LASTAB=STAB
      DTDZ=DELT
      IF(DTDZ.LT.-5.) DTDZ=-5.
      A=1./(10.+DTDZ)**2+.0025*DUDZ
      A=1./SQRT(A)
  107 STAB=1./FULGOL(A,Z0)
      IF(J.EQ.1) GO TO 108
      CHANST=ABS((LASTAB-STAB)/STAB)
      IF(CHANST.LE..01) GO TO 108
      ZOL=ZTOP/STAB
      IF(STAB.LT.0.) DUDZ=DUMB*(1.-15.*ZOL)**-.25
      IF(STAB.GT.0.) DUDZ=DUMB*(1.+4.7*ZOL)
      GO TO 100
  108 IF(STAB.LT.0.) GO TO 110
      USTAR=0.35/(ALOG(ZW/Z0)+4.7/STAB*(ZW-Z0))*U60
      USTOR=USTAR/60.
      ZHTMIX=0.5*SQRT(USTOR*STAB/F)
      ZHTMAX=0.5*USTOR/F
      IF(ZHTMIX.GT.ZHTMAX) ZHTMIX=ZHTMAX
      GO TO 115
  110 ZWPHI=(1.-15.*ZW/STAB)**.25
      Z0PHI=(1.-15.*Z0/STAB)**.25
      USTAR=0.35/(ALOG(ZW/Z0)+ALOG(((Z0PHI**2+1.)*(Z0PHI+1.)**2)/
     1((ZWPHI**2+1.)*(ZWPHI+1.)**2))+2.*(ATAN(ZWPHI)-ATAN(Z0PHI)))*U60
      USTOR=USTAR/60.
      ZHTMIN=0.5*USTOR/F
      ZHTMIX=0.5*(USTOR/F)**1.5/SQRT(-STAB)
      IF(ZHTMIX.GT.4000.) ZHTMIX=4000.
      IF((ZHTMIX.LT.ZHTMIN).OR.(J.GT.1)) ZHTMIX=ZHTMIN
  115 RETURN
      END
      SUBROUTINE STADIF(ZW,Z0,USFC,ZHIGH,DH,AKMIN,ISKLAS)
C
C-----THIS SUBROUTINE CONTROLS THE CALCULATION OF THE STABILITY
C     AND THE DIFFUSIVITIES FOR EACH HOURLY UPDATE
C
      COMMON/DIFDAT/DELTAT(30),STAB(30),ZMIX(30),USTAR(30),
     *                UZZ(201),AKZ(201)
      COMMON/TEMPHT/Z(50),ZZ(50),T(50),TT(50)
      DATA LOUT, YES, YNO /6,4HYES ,4HNO  /

      DBUG=YNO
      J=0
      N=1
  202 J=J+1
      STAB(J)=-1.E6
      DELT=DELTAT(J)
      UZZ(J+1)=USFC*ALOG(ZZ(J)/Z0)/ALOG(ZW/Z0)
      DDZ=ABS(UZZ(J+1)-UZZ(J))/(ZZ(J)-Z(J))
      ZTOP=ZZ(J)
      CALL STABIL(STAB(J),J,A,ZTOP,DDZ,DELT,USFC,ZMIX(J),USTAR(J),ZW,Z0)
      IF(J.EQ.1) ISKLAS = KLASS(A)
      USTORE=USTAR(J)/60.
  207 ZHT=Z(J)
  208 ZHT=ZHT+DH
      N=N+1
      CALL DIFFUS(STAB(J),ZHT,AKZ(N),USTAR(J))
      IF(AKZ(N).LT.AKMIN) AKZ(N)=AKMIN
      IF(ZHT.LT.ZZ(J)) GO TO 208
      IF(DBUG.EQ.YES) WRITE(LOUT,9) Z(J),ZZ(J),DELT,DDZ,A,STAB(J),
     1     USTORE,ZMIX(J)
      IF(ZZ(J).EQ.ZHIGH) GO TO 29
      GO TO 202
   29 RETURN
    9 FORMAT(1H ,F4.0,1H-,F4.0,F10.3,3X,F10.7,F10.3,F15.1,F10.2,F10.1)
      END


      FUNCTION TIMIN(T)
C
C         TIMIN CONVERTS MILITARY CLOCK TIME TO MINUTES
C
      XMIN = AMOD(T,100.)
      HRS = (T-XMIN)/100.
      TIMIN = 60.*HRS+XMIN
      RETURN
      END
      FUNCTION UNIDOT(A,B)
C
C         UNIDOT RETURNS THE DOT PRODUCT OF A UNIT VECTOR PARALLEL
C         TO  A  WITH A UNIT VECTOR PARALLEL TO  B,  I.E. THE COSINE
C         OF THE ACUTE ANGLE SEPARATING  A  AND  B.
C
      DIMENSION A(3), B(3)
      DENOM = SQRT((A(1)**2+A(2)**2+A(3)**2)*(B(1)**2+B(2)**2+B(3)**2))
      UNIDUT = 0.
      IF (DENOM.EQ.0.) RETURN
      DO 100 I = 1,3
      UNIDOT = UNIDOT+A(I)*B(I)
100   CONTINUE
      UNIDOT = UNIDOT/DENOM
      RETURN
      END



      SUBROUTINE UNITV(A,B)
C
C RETURNS B THE UNIT VECTOR PARALLEL TO VECTOR A
C
      DIMENSION A(3),B(3)
C
      X = SQRT(A(1)**2+A(2)**2+A(3)**2)
      DO 100 I = 1,3
      B(I) = A(I)/X
100   CONTINUE
      RETURN
      END



      SUBROUTINE UXYPOS
C
C-----THIS SUBROUTINE DETERMINES THE WIND SPEED AND X,Y COORDINATES
C     (LOCAL ORIGIN) AT EACH HOUR FROM THE TRAJECTORY VECTOR.
C
      COMMON/WIND/TIME(100),V(100),TH(100),NPTS
      COMMON/TRAJ/TSTART,P(2,100),ID(2,100)
      COMMON/PEVE/PE(2,24),VE(24)
C
      TE=AINT(TIME(1)*0.01)+1.
      J=IFIX(TE)
      PE(1,J)=P(1,1)
      PE(2,J)=P(2,1)
      VE(J)=V(1)
C
      DO 50 I=2,NPTS
      TI=DTIME(TIME(I))
    5 IF(TI-TE) 50,20,30
   20 J=J+1
      IF(J.GT.24) RETURN
      PE(1,J)=P(1,I)
      PE(2,J)=P(2,I)
      VE(J)=V(I)
      TE=TE+1.
      GO TO 50
   30 J=J+1
      IF(J.GT.24) RETURN
      TK=DTIME(TIME(I-1))
      TFACTR=(TI-TE)/(TI-TK)
      PE(1,J)=P(1,I)-(P(1,I)-P(1,I-1))*TFACTR
      PE(2,J)=P(2,I)-(P(2,I)-P(2,I-1))*TFACTR
      VE(J)=V(I)-(V(I)-V(I-1))*TFACTR
      TE=TE+1.
      GO TO 5
   50 CONTINUE
      RETURN
      END



      SUBROUTINE WINDRD(LUWND,KPWDAT,TITLE,JDATE)
C
C     THIS SUBROUTINE READS WIND DATA AND PUTS IT ONTO AS MANY AS  NFILE
C     SEPARATE FILES BY DATE.  THE DATES ARE NOT ORDERED.
C
      DIMENSION TITLE(20), JDATE(2)
      DIMENSION PT(24),XIMPH(24),IQ(24),JQ(24)
      DIMENSION NAM(2)
      DIMENSION IDATE(10), ICT(10)
      COMMON /DATES/ IDATE, ICT, NFILES
      DATA IBLNK/2H  /,NONO/4HNO  /
      DATA LOUT /6/
C
      NFILES = 10
C************************    ST. LOUIS ONLY   **************************
C
C     GET THE ST. LOUIS WIND DATA IN SUBROUTINE *** METIN ***
C
C     ALL CARDS BETWEEN THE LINES SHOULD BE REMOVED FOR OTHER CITIES.
C
      RETURN
C
C************************    ST. LOUIS ONLY   **************************
C
90    ISET = 0

      IOLDD = 0
      IFILE = 11
      ICOLD = 0
      IF (KPWDAT.EQ.NONO) GO TO 100
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE (LOUT,1)
      L = 0
C
100   CONTINUE
```

```
      READ (LUWND,2) NWDTE,NAM,APT,PT
      IF (NWDTE.EQ.999999) GO TO 200
      BACKSPACE LUWND
      READ (LUWND,3) IQ
      READ (LUWND,4) XIMPH
      BACKSPACE LUWND
      READ (LUWND,3) JQ
      IF (KPWDAT.EQ.NONO) GO TO 120
      L = L+1
      IF (MOD(L,51).NE.0) GO TO 110
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE (LOUT,1)
110   CONTINUE
      WRITE (LOUT,5) NWDTE,NAM,APT,PT
      WRITE (LOUT,6) XIMPH
120   CONTINUE
C
C          REPLACE MISSING DATA WITH NEGATIVE INPUTS
C
      DO 130 K = 1,24
      IF (IQ(K).EQ.IBLNK) PT(K) = -1.
      IF (JQ(K).EQ.IBLNK) XIMPH(K) = -1.
130   CONTINUE
C
      IF (NWDTE.EQ.IOLDD) GO TO 170
      IOLOD = NWDTE
      IF (ICOLD.EQ.0) GO TO 150
      DO 140 I = 1,ICOLD
      IF (IDATE(I).EQ.NWDTE) GO TO 160
140   CONTINUE
      IF (ICOLD.EQ.NFILES) GO TO 180
150   CONTINUE
      ICOLD = ICOLD+1
      IDATE(ICOLD) = NWDTE
      I = ICOLD
      ICT(ICOLD) = 0
      IFILED = IFILE + I
      REWIND IFILED
      GO TO 170
C
160   CONTINUE
      IFILED = IFILE+I
170   CONTINUE
      WRITE (IFILED) NWDTE,NAM,APT,PT,XIMPH
      ICT(IFILED-IFILE) = ICT(IFILED-IFILE) + 1
      GO TO 100
C
180   CONTINUE
      IF (ISET.GT.0) GO TO 190
      WRITE (LOUT,7) NFILES
      ISET = 1
190   WRITE (LOUT,5) NWDTE,NAM,APT,PT
      WRITE (LOUT,6) XIMPH

      GO TO 100
```

```
C
200   CONTINUE
      DO 210 I = 1,ICOLD
      IF = IFILE + I
      END FILE IF


      REWIND IF
210   CONTINUE
C
      RETURN
1     FORMAT (//1H ,50X,9HWIND DATA,//)
2     FORMAT (I6,1X,A4,A4,13X,F2.0,2X,24F2.0)
3     FORMAT (32X,24A2)
4     FORMAT (32X,24F2.0)
5     FORMAT (1H ,I6,1X,A4,A4,13X,F3.0,2HPT,24F3.0)
6     FORMAT (1H ,32X,24F3.0)
7     FORMAT (//11H MORE THAN ,I3,40H DATES HAVE BEEN FOUND IN THE WIND
     1DATA.,/,45H THE FOLLOWING ENTRIES COULD NOT BE ACCEPTED.,//)
      END


      SUBROUTINE WINDY(IWAS,LTAPE)
C
C         WINDY IS THE SECONDARY DRIVER FOR THE AIR TRAJECTORY  MODULE
C
C     MODIFIED BY F.W. LURMANN     (12.9.77)
C
      DIMENSION KOLD(2),KNEW(2)
      DIMENSION DDD(3)
      DIMENSION AZV(12)
      DIMENSION ID(2),IDD(2)
      COMMON/INPUTS/  TITLE(20),          JDATE(10),          NCURV
      COMMON /AIRQAL/ NEEDAQ,    NOSPEC,   CONAM(10),   CON(24,25,10)
      COMMON /CNTROL/ KSTOP, TSUN
      COMMON /DATES/  IDATE(10),  ICT(10),  NFILES
      COMMON  /GRID/  X1, X2, Y1, Y2, NX, NY, DELX, DELY, DELT
      COMMON /KZINPT/ NEEDKZ,             KZPRTX,             NUMSD,
     1                TIMESD(3),          TEMPSD(50,3),       ZELVSD(50,3)
     2      ,         TEMPSF(24),         NPTSD(3)        ,   NKZDAT
      COMMON /ORIGIN/ UTMXOR, UTMYOR
      COMMON /REUSE/  KPSTAT,  KPWDAT,  CONVRT,  KXTRA, KPWIND
      COMMON /TRAJ/   TSTART, P(2,100), IDXX(2,100)
      COMMON /WDATA/  NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
      COMMON /WHERE/  RLAT,           RLONG,                THZONE
      COMMON /WIND/   T(100), V(100), TH(100), NPTS
      COMMON /WINFLD/ AWDATA(52,25) ,DISAVE(25),   IDSAVE(25)
      INTEGER ENTRAJ
C
      DATA DALITE/4HDAYL/,IBLNK/1H /,NONO/4HNO  /
      DATA NIT/3/
      DATA RTD/57.29578/, NEG/2HNO/
      DATA IDATE  /10*0/
      DATA LIN,LOUT,LPUNCH /3,6,1/
      DATA KYES/4HYES /
      ENTRAJ = NONO
      IF (IWAS.GT.0) GO TO 130
C
   95 CONTINUE
C
      READ(LIN,9) KPSTAT,KPWDAT
      READ(LIN,9) KXTRA
      READ(LIN,11) CONVRT
      IF (CONVRT.LE.0.) CONVRT = 1.0
      READ(LIN,11)  UTMXOR,  UTMYOR

      READ(LIN,11)  DELX,  DELY
      READ(LIN,31)  NX, NY
      READ(LIN,11)  RMIN,  RMAX
      RMIN = RMIN**2
      RMAX = RMAX**2
      CALL SETIN
      IF (KPSTAT.EQ.NONO) GO TO 120
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,1)
      DO 110 I = 1,NUMSTA
      IF (MOD(I,52).NE.0) GO TO 100
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,1)
100   CONTINUE
      WRITE(LOUT,2) ISTANS(1,I),ISTANS(2,I),SSTAN(1,I),SSTAN(2,I)
110   CONTINUE
120   CONTINUE
      IF(ENTRAJ.EQ.KYES)  GO TO 333
C
C     READ WIND DATA
      CALL WINDRD(LTAPE,KPWDAT,TITLE,JDATE)
      IWAS = 1
C
130   CONTINUE
C
C     READ CASE DATA
C
      ICASES = 1
      IK = 0
      READ(LIN,8) IDT,XLCL
      TSUN = 0.
      READ(LIN,9) STIME
      IF (STIME.EQ.DALITE) TSUN = 0100.
      ILCL = IFIX(XLCL)
      IST = IFIX(XLCL-TSUN)
      READ(LIN,10) ID,XS,YS ←IF(ID(1).NE.IBLK)GOTO 132
      XS = XS - UTMXOR
      YS = YS - UTMYOR
132   READ(LIN,11) TTOTAL
      READ(LIN,11) DTSEG
      ITN = IFIX(60.*DTSEG)
      NN = IFIX((TTOTAL+0.9*DTSEG)/DTSEG)
      READ(LIN,11) AZ,VEL
      READ(LIN,12) ITF
      READ(LIN,12) IRF
      READ(LIN,12) NC
C
      IF(IDT.EQ.IDATE(1))  GO TO 134
C       GET THE METEORELOGICAL AND AIR QUALITY DATA FOR ST. LOUIS.
      CALL METIN (IDT,LTAPE)
  134 CONTINUE
C
C      IDT    DATE YEAR-MONTH-DAY
C      ID     STATION IDENTIFICATION (OR BLANK)
C      XS     X COORDINATE OF START
C      YS     Y COORDINATE OF START
C      ILCL   STARTING TIME OF DAY - 24 HOUR CLOCK - LOCAL TIME
C      IST    STARTING TIME OF DAY - 24 HOUR CLOCK - STANDARD TIME
C      ITN    TIME BETWEEN MOVES
C      NN     NUMBER OF MOVES TO BE MADE
C      AZ     STARTING AZIMUTH(IF ANY)
C      VEL    STARTING VELOCITY (IF ANY)

C      ITF    IF POS MOVE FORWARD IN TIME, IF NEG MOVE BACKWARD
C      IRF    IF 1 CALCULATE INVERSE DISTANCE, IF 2 CALCULATE INVERSE SQU
C      NC     NUMBER OF CLOSEST STATIONS TO TRY TO USE
C
C
C      IF STATION ID IS BLANK DO COORDINATES CORRESPOND TO A KNOWN STATIO
C
       IF (ID(1).NE.IBLNK) GO TO 150
       DO 140 I = 1,NUMSTA
       IF (XS.NE.SSTAN(1,I)) GO TO 140
       IF (YS.NE.SSTAN(2,I)) GO TO 140
       ID(1) = ISTANS(1,I)
       ID(2) = ISTANS(2,I)
       GO TO 150
 140   CONTINUE
 150   CONTINUE
C
C IF COORDINATES ARE ZERO LOOK UP STATION AND SUPPLY COORDINATES
       IF (XS.NE.0..OR.YS.NE.0.) GO TO 170
       DO 160 I = 1,NUMSTA
       IF (ID(1).NE.ISTANS(1,I)) GO TO 160
       IF (ID(2).NE.ISTANS(2,I)) GO TO 160
       XS = SSTAN(1,I)
       YS = SSTAN(2,I)
       GO TO 170
 160   CONTINUE
 170   CONTINUE
C
C      PRINT THE INITIAL INPUTS
C
       CALL NEWPAG(TITLE,0,JDATE)
       WRITE(LOUT,13) IDT,ILCL
       IF (TSUN.EQ.0.) WRITE(LOUT,14)
       IF (TSUN.EQ.0100.) WRITE(LOUT,15)
      WRITE(LOUT,16) XS,YS
      IF (ID(1).EQ.IBLNK) WRITE(LOUT,17)
      IF (ID(1).NE.IBLNK) WRITE(LOUT,18) ID
      WRITE(LOUT,19)
      IF (VEL.GT.0.) WRITE(LOUT,20) AZ,VEL
      WRITE(LOUT,21)
      IF (ITF.GE.0) WRITE(LOUT,22)
      IF (ITF.LT.0) WRITE(LOUT,23)
      WRITE(LOUT,24) TTOTAL
      WRITE(LOUT,25) DTSEG
      WRITE(LOUT,26) NC
      IF (IRF.EQ.1) WRITE(LOUT,27)
      IF (IRF.EQ.2) WRITE(LOUT,28)
      IF(KXTRA.EQ.KYES)  WRITE(LOUT,33)
C
      KOLD(1) = IST
      KOLD(2) = IDT
      KNEW(1) = IST
      KNEW(2) = IDT
      IDD(1) = ID(1)
      IDD(2) = ID(2)
      IF (ITF.LT.0) CALL DATE(KOLD,ITN,ITF,KNEW)
      IF (VEL.GT.0.) GO TO 190
      CALL GETAZV(KNEW,IDD,XS,YS,NC,AZV,NGD,DDD)
      IF (NGD.LE.0) GO TO 180
      CALL AZVDIS(NGD,AZV,DDD,AZ,VEL,IRF)
      VEL = VEL*CONVRT
      GO TO 190

180   CONTINUE
      IK = IK-1
      WRITE(LOUT,3)
      GO TO 280
C
190   CONTINUE
      IK = 1
      P(1,1) = XS
      P(2,1) = YS
      NTM = 0
      XMUL = DTSEG
      IF (ITF.GT.0) GO TO 200
      XMUL = -DTSEG
      T(1) = FLOAT(KOLD(1))
      IK = 2
      GO TO 250
200   CONTINUE
      IF (KXTRA.EQ.KYES) WRITE(LOUT,4) KOLD,XS,YS
      IF (ITF.GT.0) GO TO 210
      KOLD(1) = KNEW(1)
      KOLD(2) = KNEW(2)
210   CONTINUE
C
C     APPLY MOVE NOW TO GET NEW POINT
      ANG = 90.-AZ
      IF (ANG.LT.0.) ANG = ANG+360.
      ANGUSE = ANG/57.29577951
      T(IK) = FLOAT(KOLD(1))
      V(IK) = VEL
      TH(IK) = ANGUSE
      IF (KXTRA.EQ.KYES) WRITE(LOUT,5) KOLD,VEL,ANG,(AZV(I),I=4,12,4)
      XNEW = XS+XMUL*COS(ANGUSE)*VEL
      YNEW = YS+XMUL*SIN(ANGUSE)*VEL
      CALL BARIER(XNEW,YNEW,XS,YS,OK)
      IF (OK.LT.0.) GO TO 320
      CALL EDGE(XNEW,YNEW,ITF,IEDGE)
      IF (IEDGE) 330,230,220
220   CONTINUE
      IF (NTM.GE.NN) GO TO 280
      NN = NTM+1
230   CONTINUE
      XS = XNEW
      YS = YNEW
C
C     LAST MOVE CHECK
      IF (NTM.GE.NN) GO TO 280
C
      IK = IK+1
C
C     CHANGE DATE
      CALL DATE(KOLD,ITN,ITF,KNEW)
      IF (ITF.LT.0) GO TO 240
      KOLD(1) = KNEW(1)
      KOLD(2) = KNEW(2)
240   CONTINUE
      IDD(1) = IBLNK
      IDD(2) = IBLNK
      CALL GETAZV(KNEW,IDD,XS,YS,NC,AZV,NGD,DDD)
      IF (NGD.LE.0) GO TO 180
      CALL AZVDIS(NGD,AZV,DDD,AZ,VEL,IRF)
      VEL = VEL*CONVRT
C

C     WE JUST GOT A NEW AZIMUTH AND VELOCITY
C     IF WE ARE MOVING BACKWARD IN TIME, THERE FOLLOW  NIT  ITERATIONS
C     TO REFINE THE AZIMUTH AND VELOCITY.
C     (NIT  IS SET IN A DATA STATEMENT ABOVE.)
C
      IF(ITF.GT.0)  GO TO 270
C
C
250   CONTINUE
      IF (NIT.EQ.0) GO TO 270
      DO 260 IT = 1,NIT
      ANG1 = 90.-AZ
      IF (ANG1.LT.0.) ANG1 = ANG1+360.
      ANGYU = ANG1/57.29577951
      XNUT = XS+XMUL*COS(ANGYU)*VEL
      YNUT = YS+XMUL*SIN(ANGYU)*VEL
      CALL BARIER(XNUT,YNUT,XS,YS,OK)
      IF (OK.LT.0) GO TO 310
      IDD(1) = IBLNK
      IDD(2) = IBLNK
      CALL GETAZV(KNEW,IDD,XNUT,YNUT,NC,AZV,NGD,DDD)
      IF (NGD.LE.0) GO TO 180
      CALL AZVDIS(NGD,AZV,DDD,AZ,VEL,IRF)
      VEL = VEL*CONVRT
260   CONTINUE
270   CONTINUE
      NTM = NTM+1
      GO TO 200
C
280   CONTINUE
      NPTS = IK
C
      IF (ITF.GT.0) GO TO 300
C
C     REVERSE THE ORDER OF THE TIME, VELOCITY, AND ANGLE LISTS
C
      NXY = (IK-MOD(IK,2))/2
      DO 290 K = 1,NXY
      J = IK-K+1
      TT = T(K)
      T(K) = T(J)
      T(J) = TT
      VV = V(K)
      V(K) = V(J)
      V(J) = VV
      AA = TH(K)
      TH(K) = TH(J)
      TH(J) = AA
290   CONTINUE
      V(IK) = V(IK-1)
      TH(IK) = TH(IK-1)
      P(1,1) = XS
      P(2,1) = YS
300   CONTINUE
      IF (KPWIND.EQ.NEG) GO TO 308
      WRITE(LPUNCH,30)  TITLE
      A = P(1,1) + UTMXOR
      B = P(2,1) + UTMYOR
      WRITE(LPUNCH,47)  A, B
      IF(TSUN.EQ.0.)  WRITE(LPUNCH,48)
      IF(TSUN.GT.0.)  WRITE(LPUNCH,49)
      DO 305 I = 1,NPTS

      A = TH(I)*RTD
      B = TIMIN(T(I))
      WRITE(LPUNCH,32)  B,  V(I),  A
  305 CONTINUE
      A = -100.
      WRITE(LPUNCH,32)  A
308   CONTINUE
C
      GO TO 500
C PRINT MESSAGE
```

```
310   CONTINUE
      XNEW = XNUT
      YNEW = YNUT
320   CONTINUE
      IF(ITF.LT.0) IK = IK - 1
      WRITE(LOUT,6) XS,YS,XNEW,YNEW
      GO TO 280
330   CONTINUE
      IK = IK-1
      WRITE(LOUT,7)
      GO TO 280
 332  CONTINUE
C
C
C         FOR THE UNIVAC (OR IBM) REPLACE   *ENTRY TRAJEC*  WITH
C         *ENTRY TRAJEC(IWAS,KPWIND,LTAPE)*
C
      ENTRY TRAJEC
C
      IF(IWAS.GT. 0)  GO TO 335
      ENTRAJ = KYES
      GO TO 95
  333 CONTINUE
      IWAS = 1
  335 CONTINUE
C
C         READ INPUTS FOR INDIVIDUAL CASE
C
      READ(LIN,8)  IDT
      READ(LIN,12) IRF
      READ(LIN,12) NC
C
      P(1,1) = P(1,1) - UTMXOR
      P(2,1) = P(2,1) - UTMYOR
      IF(IDT.EQ.IDATE(1))  GO TO 350
C
C        GET THE METEORELOGICAL AND AIR QUALITY DATA FOR ST. LOUIS.
      IF(NEEDAQ.NE.KYES)  GO TO 350
      CALL METIN (IDT,LTAPE)
  350 CONTINUE
C
      ILCL = IFIX(T(1))
C     PRINT THE INITIAL INPUT
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,13) IDT,ILCL
      IF (TSUN.EQ.0.) WRITE(LOUT,14)
      IF (TSUN.EQ.0100.) WRITE(LOUT,15)
      WRITE(LOUT,16)  P(1,1), P(2,1)
      WRITE(LOUT,26) NC
      IF (IRF.EQ.1) WRITE(LOUT,27)
      IF (IRF.EQ.2) WRITE(LOUT,28)
      DO 360 J = 1,NPTS

      V(J) = V(J)*CONVRT
 360  CONTINUE
```

```
C
 500  CONTINUE
C
      CALL CROSIT
      IF(KSTOP.GT.0)  GO TO 540
C
      CALL EMETIN
C
C     CONVERT TIME FROM MINUTES TO 2400 - HOUR CLOCK AND CONVERT
C     WIND SPEEDS FROM KILOMETERS/MINUTE TO METERS/SECOND
      DT = T(2) - T(1)  + .001
      T(1) = TSTART
      V(1) = V(1)*16.666667
      DO 510 J = 2,NPTS
      IF(J.GT.100)  GO TO 515
      DTNXT = T(J+1) - T(J) + .001
      T(J) = CLOCKT( T(J-1) , DT)
      DT = DTNXT
      V(J) = V(J)*16.666667
 510  CONTINUE
  515 CONTINUE
C
      IF(NEEDAQ.NE.KYES)  GO TO 530
C
      CALL NEWPAG(TITLE,0,JDATE)
      KNEW(2) = IDT
      DO 520 J = 1,NPTS
      KNEW(1) = IFIX(T(J))
      CALL WSECLS (P(1,J),P(2,J),DISAVE,IDSAVE)
      CALL GETCON (KNEW,NC,IRF)
      IF(J.EQ.NPTS)  GO TO 520
      IF(MOD(J,2).EQ. 0)  CALL NEWPAG(TITLE,0,JDATE)
  520 CONTINUE
  530 CONTINUE
C
      IF(NEEDKZ.NE.KYES)  GO TO 540
      CALL EDDY(KPWIND)
  540 CONTINUE
      RETURN
C
C
1     FORMAT (//10X,25H STATIONS AND COORDINATES,/)
2     FORMAT (1H ,10X,2A4,2F10.3)
3     FORMAT (26H0THERE ARE NO CLOSE POINTS)
4     FORMAT (1H ,I6,4X,I6,4X,2F6.2)
5     FORMAT (1H+,40X,I6,4X,I6,4X,2F8.2,2X,3(2X,A4))
6     FORMAT (27H0POINT MOVED ACROSS BARRIER,2(4X,2F10.2)//)
7     FORMAT (/42H0TRAJECTORY BACKED MORE THAN 15 KM OF GRID   //)
8     FORMAT (40X,I6,4X,F4.0)
9     FORMAT (40X,A4,6X,A4)
10    FORMAT (40X,2A4,2X,2F10.0)
11    FORMAT (40X,2F10.0)
12    FORMAT (40X,I2)
13    FORMAT (///,33H  AIR TRAJECTORY CALCULATION DATA ,///
     1  47H     START DATE AND LOCAL TIME AT INITIAL POINT,13X,I6,4X,I4)
```

```
14     FORMAT (1H+,73X,20H HRS (STANDARD TIME))
15     FORMAT (1H+,73X,20H HRS (DAYLIGHT TIME))
16     FORMAT ( /,30H     LOCATION OF INITIAL POINT,30X,F6.2,4X,F6.2,3X)
17     FORMAT (1H+,78X,30H(NOT AT A MEASUREMENT STATION))
18     FORMAT (1H+,78X,1H(,A4,A4,23H  MEASUREMENT STATION)    )

19     FORMAT (/,
      1  56H     PRESCRIBED STARTING AZIMUTH AND VELOCITY (OPTIONAL) )
20     FORMAT (1H+,59X,F6.2,4X,F6.2,  6H KM/HR  )
21     FORMAT ( /,40H     DIRECTION OF TRAJECTORY DEVELOPMENT )
22     FORMAT (1H+,59X,7HFORWARD )
23     FORMAT (1H+,59X,8HBACKWARD )
24     FORMAT ( /,32H     DESIRED TRAJECTORY DURATION,28X,F5.1,4H HRS )
25     FORMAT ( /,30H     TRAJECTORY SEGMENT LENGTH,30X,F5.1,4H HRS )
26     FORMAT ( /,42H     MEASUREMENT INTERPOLATION SCHEME     ,18X,I1)
27     FORMAT (1H+,60X,39H  CLOSEST STATIONS WITH  1/R  WEIGHTING ,//)
28     FORMAT (1H+,60X,42H  CLOSEST STATIONS WITH  1/R**2  WEIGHTING ,//)
30     FORMAT (20A4)
31     FORMAT (40X,2I10)
32     FORMAT (40X,3F10.3)
33     FORMAT (1H0,//,17H OLD TIME - DATE ,7X,1HX,5X,1HY,10X,
      1 16HNEW TIME - DATE ,6X,16HVELOCITY AZIMUTH  ,4X,8HSTATIONS ,/)
47     FORMAT (23HSTART LOCATION (COORDS),17X,2F10.2)
48     FORMAT (34HIS LOCAL TIME STANDARD OR DAYLIGHT,6X,8HSTANDARD)
49     FORMAT (34HIS LOCAL TIME STANDARD OR DAYLIGHT,6X,8HDAYLIGHT)
       END


      SUBROUTINE WSECLS(X,Y,DIS,IDIS)
C
C         THIS SUBROUTINE CALCULATES SQUARED DISTANCES FROM A GIVEN
C         POINT TO EACH OF A SET OF POINTS AND THEN ORDERS THE DISTANCES
C         OUT TO A MAXIMUM VALUE.
C
C         X          X COORDINATE OF GIVEN POINT
C         Y          Y COORDINATE OF GIVEN POINT
C         DIS        THE SQUARE OF THE DISTANCE FROM THE GIVEN POINT TO
C                    ALL THE POINTS IN THE POINT SET.  THOSE VALUES OF
C                    DIS LESS THAN RMAX ARE ARRANGED IN INCREASING
C                    ORDER AT THE TOP OF THE TABLE.
C         IDIS       THE NUMBER OF THE POINT IN THE POINT SET
C                    CORRESPONDING TO THE DIS ARRAY
C
       DIMENSION DIS(2), IDIS(2)
       COMMON /WDATA/ NUMSTA, SSTAN(2,25),  ISTANS(2,25),  RMIN,  RMAX
C
       DO 100 I = 1,NUMSTA
       IDIS(I) = I
       DIS(I) = (SSTAN(1,I)-X) **2+(SSTAN(2,I)-Y) **2
100    CONTINUE
C
       DO 110 N = 1,NUMSTA
       M = NUMSTA+1-N
       CALL FMINF(DIS(N),M,FMIN,NMIN)
       NMIN = NMIN+N-1                                                  MM  3282
       IF (FMIN.GT.RMAX) GO TO 120                                      MM  3283
       J = IDIS(N)                                                      MM  3284
       IDIS(N) = IDIS(NMIN)                                             MM  3285
       IDIS(NMIN) = J                                                   MM  3286
       XX = DIS(N)                                                      MM  3287
       DIS(N) = FMIN                                                    MM  3288
       DIS(NMIN) = XX                                                   MM  3289
 110   CONTINUE                                                         MM  3290
C                                                                       MM  3291
 120   RETURN                                                           MM  3292
       END                                                              MM  3293



       BLOCK DATA                                                       MM  3294
C                                                                       MM  3295
C          BLOCK DATA PROGRAM NUMBER ONE                                MM  3296
C          VARIABLE SIZE GRID ID NUMBERS  FOR ST. LOUIS                 MM  3297
       INTEGER A1. A2, A3, A4, A5, A6, A7, A8                           MM  3298
       DIMENSION   A1(341),  A2(370),  A3(439) , A4(441),               MM  3299
      1            A5(319),  A6(315),  A7(308),  A8(248)                MM  3300
       COMMON /VGRID/  ITGRID,      IVG(100,100)                        MM  3301
       EQUIVALENCE (A1(1),IVG(1,1))                                     MM  3302
       EQUIVALENCE (A2(1),IVG(42,4))                                    MM  3303
       EQUIVALENCE (A3(1),IVG(12,8))                                    MM  3304
       EQUIVALENCE (A4(1),IVG(51,12))                                   MM  3305
       EQUIVALENCE (A5(1),IVG(92,16))                                   MM  3306
       EQUIVALENCE (A6(1),IVG(11,20))                                   MM  3307
       EQUIVALENCE (A7(1),IVG(26,23))                                   MM  3308
       EQUIVALENCE (A8(1),IVG(34,26))                                   MM  3309
C                                                                       MM  3310
       DATA ITGRID /4HYES /                                             MM  3311
C                                                                       MM  3312
       DATA A1/                                                         MM  3313
                                                                        MM  3314
      1     64, 4*  64,     72, 4*  72,     79, 4*  79,     96, 9*  96,MM  3315
      2    147, 4* 147,    181, 4* 181,    214, 4* 214,    291, 4* 291,MM  3316
      3    348,    395,    424,    467, 1* 467,    528, 4* 528,    695,MM  3317
      4 4* 695,    878, 4* 878,   1056, 4*1056,   1205, 4*1205,   1373,MM  3318
      5 4*1373,   2335, 2*2335,   2361, 1*2361,   1582, 4*1582,   1616,MM  3319
      6 4*1616,   1633, 4*1633,     64, 4*  64,     72, 4*  72,     79,MM  3320
      7 4*  79,     96, 9*  96,    147, 4* 147,    181, 4* 181,    214,MM  3321
      8 4* 214,    291, 4* 291,    349,    396,    425,    467, 1* 467,MM  3322
      9    528, 4* 528,    695, 4* 695,    878, 4* 878,   1056, 4*1056,MM  3323
      1   1205, 4*1205,   1373, 4*1373,   2335, 2*2335,   2361, 1*2361,MM  3324
      2   1582, 4*1582,   1616, 4*1616,   1633, 4*1633,     64, 4*  64,MM  3325
      3     72, 4*  72,     79, 4*  79,     96, 9*  96,    147, 4* 147,MM  3326
      4    181, 4* 181,    214, 4* 214,    291, 4* 291,    350,    397,MM  3327
      5    426,    468,    496,    528, 4* 528,    695, 4* 695,    878,MM  3328
      6 4* 878,   1056, 4*1056,   1205, 4*1205,   1373, 4*1373,   2335,MM  3329
      7 2*2335,   2362,   2368,   1582, 4*1582,   1616, 4*1616,   1633,MM  3330
      8 4*1633,     64, 4*  64,     72, 4*  72,     79, 4*  79,     96,MM  3331
      9 9*  96,    147, 4* 147,    181, 4* 181,    214, 4* 214,    291/MM  3332
       DATA A2/
                                                                        MM  3333
```

```
1    4* 291,     351,     398,     427,     469,     497,     528, 4* 528,MM  3334
2       695, 4* 695,     678, 4* 878,    1056, 4*1056,    1205, 4*1205,MM  3335
3      1373, 4*1373,    2336, 1*2336,    2347,    2363, 1*2363,    1582,MM  3336
4    4*1582,    1616, 4*1616,    1633, 4*1633,      64, 4*  64,      72,MM  3337
5    4*  72,      79, 4*  79,      96, 9*  96,     147, 4* 147,     181,MM  3338
6    4* 181,     214, 4* 214,     291, 4* 291,     352,     399,     428,MM  3339
7       470,     498,     528, 4* 528,     695, 4* 695,     878, 4* 878,MM  3340
8      1056, 4*1056,    1205, 4*1205,    1373, 4*1373,    2336, 1*2336,MM  3341
9      2348,    2363, 1*2363,    1582, 4*1582,    1616, 4*1616,    1633,MM  3342
1    4*1633,      65, 9*  65,      80, 4*  80,      97, 4*  97,     126,MM  3343
2    4* 126,     148, 4* 148,     182, 4* 182,     215, 4* 215,     292,MM  3344
3    4* 292,     353,     400,     429,     471,     499,     529, 4* 529,MM  3345
4       696, 4* 696,     879, 4* 879,    1057, 9*1057,    1374, 4*1374,MM  3346
5      1480, 9*1480,    1617, 9*1617,      65, 9*  65,      80, 4*  80,MM  3347
6        97, 4*  97,     126, 4* 126,     148, 4* 148,     182, 4* 182,MM  3348
7       215, 4* 215,     292, 4* 292,     354, 3* 354,     500,     529,MM  3349
8    4* 529,     696, 4* 696,     879, 4* 879,    1057, 9*1057,    1374,MM  3350
9    4*1374,    1480, 9*1480,    1617, 9*1617,      65, 9*  65,      80/MM  3351
 DATA A3/                                                             MM  3352
1    4*  80,      97, 4*  97,     126, 4* 126,     148, 4* 148,     182,MM  3353
2    4* 182,     215, 4* 215,     292, 4* 292,     354, 3* 354,     501,MM  3354
3       529, 4* 529,     696, 4* 696,     879, 4* 879,    1057, 9*1057,MM  3355
4      1374, 4*1374,    1480, 9*1480,    1617, 9*1617,      65, 9*  65,MM  3356
5        80, 4*  80,      97, 4*  97,     126, 4* 126,     148, 4* 148,MM  3357
6       182, 4* 182,     215, 4* 215,     292, 4* 292,     354, 3* 354,MM  3358
7       502,     529, 4* 529,     696, 4* 696,     879, 4* 879,    1057,MM  3359
8    9*1057,    1374, 4*1374,    1480, 9*1480,    1617, 9*1617,      65,MM  3360
9    9*  65,      80, 4*  80,      97, 4*  97,     126, 4* 126,     148,MM  3361
1    4* 148,     182, 4* 182,     215, 4* 215,     292, 4* 292,     354,MM  3362
2    3* 354,     503,     529, 4* 529,     696, 4* 696,     879, 4* 879,MM  3363
3      1057, 9*1057,    1374, 4*1374,    1480, 9*1480,    1617, 9*1617,MM  3364
4        65, 9*  65,      81, 4*  81,      98, 4*  98,     127, 4* 127,MM  3365
5       149, 4* 149,     183, 4* 183,     216, 4* 216,     293, 4* 293,MM  3366
6       355, 4* 355,     530, 4* 530,     697, 9* 697,    1057, 9*1057,MM  3367
7      1375, 4*1375,    1480, 9*1480,    1617, 9*1617,      65, 9*  65,MM  3368
8        81, 4*  81,      98, 4*  98,     127, 4* 127,     149, 4* 149,MM  3369
9       183, 4* 183,     216, 4* 216,     293, 4* 293,     355, 4* 355/MM  3370
 DATA A4/                                                             MM  3371
1       530, 4* 530,     697, 9* 697,    1057, 9*1057,    1375, 4*1375,MM  3372
2      1480, 9*1480,    1617, 9*1617,      65, 9*  65,      81, 4*  81,MM  3373
3        98, 4*  98,     127, 4* 127,     149, 4* 149,     183, 4* 183,MM  3374
4       216, 4* 216,     293, 4* 293,     355, 4* 355,     530, 4* 530,MM  3375
5       697, 9* 697,    1057, 9*1057,    1375, 4*1375,    1480, 9*1480,MM  3376
6      1617, 9*1617,      65, 9*  65,      81, 4*  81,      98, 4*  98,MM  3377
7       127, 4* 127,     149, 4* 149,     183, 4* 183,     216, 4* 216,MM  3378
8       293, 4* 293,     355, 4* 355,     530, 4* 530,     697, 9* 697,MM  3379
9      1057, 9*1057,    1375, 4*1375,    1480, 9*1480,    1617, 9*1617,MM  3380
1        65, 9*  65,      81, 4*  81,      98, 4*  98,     127, 4* 127,MM  3381
2       149, 4* 149,     183, 4* 183,     216, 4* 216,     293, 4* 293,MM  3382
3       355, 4* 355,     530, 4* 530,     697, 9* 697,    1057, 9*1057,MM  3383
4      1375, 4*1375,    1480, 9*1480,    1617, 9*1617,      66, 9*  66,MM  3384
5        82, 4*  82,      99, 4*  99,     128, 4* 128,     150, 4* 150,MM  3385
6       184, 4* 184,    2088, 2*2088,    2129, 2*2129,    2151, 2*2151,MM  3386
7      2188, 2*2188,    2203, 2*2203,    2232, 1*2232,    2246, 2*2246,MM  3387
8       697, 9* 697,    1058, 1*1058,    1116,    1145,    1174,    1206,MM  3388
```

```
9   4*1206,     1376,  4*1376,      1481,  4*1481,      1583,  4*1583,      1618/MM  3389
 DATA A5/                                                                     MM  3390
1   4*1618,     1534,  4*1634,        66,  9*  66,        82,  4*  82,        99,MM  3391
2   4*  99,      128,  4* 128,       150,  4* 150,       184,  4* 184,      2088,MM  3392
3   2*2088,     2129,  2*2129,      2151,  2*2151,      2188,  2*2188,      2203,MM  3393
4   2*2203,     2232,  1*2232,      2246,  2*2246,       697,  9* 697,      1058,MM  3394
5   1*1058,     1117,    1146,      1175,    1206,  4*1206,    1376,  4*1376,MM  3395
6      1481,  4*1481,    1583,  4*1583,    1618,  4*1618,    1634,  4*1634,MM  3396
7        66,  9*  66,      82,  4*  82,      99,  4*  99,     128,  4* 128,MM  3397
8       150,  4* 150,     184,  4* 184,    2088,  2*2088,    2129,  2*2129,MM  3398
9      2151,  2*2151,    2188,  2*2188,    2203,  2*2203,    2233,    2235,MM  3399
1      2246,  2*2246,     697,  9* 697,    1059,  2*1059,    1147,    1176,MM  3400
2      1206,  4*1206,    1376,  4*1376,    1481,  4*1481,    1583,  4*1583,MM  3401
3      1618,  4*1618,    1634,  4*1634,      66,  9*  66,      82,  4*  82,MM  3402
4        99,  4*  99,     128,  4* 128,     150,  4* 150,     184,  4* 184,MM  3403
5      2089,  1*2089,    2111,  1*2111,    2131,  1*2131,    2152,  1*2152,MM  3404
6      2171,  1*2171,    2190,  1*2190,    2204,  1*2204,    2218,  1*2218,MM  3405
7      2236,  1*2236,    2247,  1*2247,     697,  9* 697,    1059,  2*1059,MM  3406
8      1148,    1177,    1206,  4*1206,    1376,  4*1376,    1481,  4*1481,MM  3407
9      1583,  4*1583,    1618,  4*1618,    1634,  4*1634,      66,  9*  66/MM  3408
 DATA A6/                                                                MM  3409
1        82,  4*  82,      99,  4*  99,     128,  4* 128,     150,  4* 150,MM  3410
2       184,  4* 184,    2089,  1*2089,    2111,  1*2111,    2131,  1*2131,MM  3411
3      2152,  1*2152,    2171,  1*2171,    2190,  1*2190,    2204,  1*2204,MM  3412
4      2218,  1*2218,    2236,  1*2236,    2247,  1*2247,     697,  9* 697,MM  3413
5      1059,  2*1059,    1149,    1178,    1206,  4*1206,    1376,  4*1376,MM  3414
6      1481,  4*1481,    1583,  4*1583,    1618,  4*1618,    1634,  4*1634,MM  3415
7        66,  9*  66,    2026,  4*2026,    2028,  4*2028,     129,  4* 129,MM  3416
8       151,  4* 151,     185,  4* 185,    2090,  3*2090,    2132,  1*2132,MM  3417
9      2153,  1*2153,    2172,  1*2172,    2191,  1*2191,    2205,  1*2205,MM  3418
1      2219,  1*2219,    2237,  3*2237,     698,  4* 698,     880,  4* 880,MM  3419
2      1060,  4*1060,    1207,  4*1207,    1377,  4*1377,    1482,  4*1482,MM  3420
3      1584,  9*1584,    1635,  4*1635,      66,  9*  66,    2026,  4*2026,MM  3421
4      2028,  4*2028,     129,  4* 129,     151,  4* 151,     185,  4* 185,MM  3422
5      2090,  3*2090,    2132,  1*2132,    2153,  1*2153,    2172,  1*2172,MM  3423
6      2191,  1*2191,    2205,  1*2205,    2219,  1*2219,    2237,  3*2237,MM  3424
7       698,  4* 698,     880,  4* 880,    1060,  4*1060,    1207,  4*1207,MM  3425
8      1377,  4*1377,    1482,  4*1482,    1584,  9*1584,    1635,  4*1635,MM  3426
9        66,  9*  66,    2026,  4*2026,    2028,  4*2028,     129,  4* 129/MM  3427
 DATA A7/                                                                MM  3428
1       151,  4* 151,     185,  4* 185,    2090,  3*2090,    2133,  1*2133,MM  3429
2      2154,  1*2154,    2173,  1*2173,    2192,  1*2192,    2206,  1*2206,MM  3430
3      2220,  1*2220,    2237,  3*2237,     698,  4* 698,     880,  4* 880,MM  3431
4      1060,  4*1060,    1207,  4*1207,    1377,  4*1377,    1482,  4*1482,MM  3432
5      1584,  9*1584,    1635,  4*1635,      66,  9*  66,    2026,  4*2026,MM  3433
6      2028,  4*2028,     129,  4* 129,     151,  4* 151,     185,  4* 185,MM  3434
7      2090,  3*2090,    2133,  1*2133,    2154,  1*2154,    2173,  1*2173,MM  3435
8      2192,  1*2192,    2206,  1*2206,    2220,  1*2220,    2237,  3*2237,MM  3436
9       698,  4* 698,     880,  4* 880,    1060,  4*1060,    1207,  4*1207,MM  3437
1      1377,  4*1377,    1482,  4*1482,    1584,  9*1584,    1635,  4*1635,MM  3438
2        66,  9*  66,    2026,  4*2026,    2028,  4*2028,     129,  4* 129,MM  3439
3       151,  4* 151,     185,  4* 185,    2091,  1*2091,    2112,  1*2112,MM  3440
4      2134,  1*2134,    2155,  1*2155,    2174,  1*2174,    2193,  1*2193,MM  3441
5      2207,  1*2207,    2221,  1*2221,    2238,  1*2238,    2248,  1*2248,MM  3442
6       698,  4* 698,     880,  4* 880,    1060,  4*1060,    1207,  4*1207,MM  3443
```

```
     7     1377, 4*1377,   1482, 4*1482,   1584, 9*1584,   1635, 4*1635,
     8       67, 9*  67,   2027, 4*2027,   2029, 4*2029,    130, 4* 130,
     9     2039, 1*2039,   2048, 1*2048,   2057, 1*2057,   2065, 1*2065/
      DATA A8/
     1     2075, 1*2075,   2091, 1*2091,   2112, 1*2112,   2134, 1*2134,
     2     2155, 1*2155,   2174, 1*2174,   2193, 1*2193,   2207, 1*2207,
     3     2221, 1*2221,   2238, 1*2238,   2248, 1*2248,    699, 4* 699,
     4      881, 1* 881,    948,    990, 1* 990,   1061, 4*1061,   1208,
     5   9*1208,   1483, 4*1483,   1584, 9*1584,   1636, 4*1636,     67,
     6   9*  67,   2027, 4*2027,   2029, 4*2029,    130, 4* 130,   2039,
     7   1*2039,   2048, 1*2048,   2057, 1*2057,   2065, 1*2065,   2075,
     8   1*2075,   2092, 1*2092,   2113, 1*2113,   2135, 1*2135,   2156,
     9   1*2156,   2175, 1*2175,   2194, 1*2194,   2208, 1*2208,   2222,
     1   1*2222,   2239, 1*2239,   2249, 1*2249,    699, 4* 699,    881,
     2   1* 881,    949,    990, 1* 990,   1061, 4*1061,   1208, 9*1208,
     3     1483, 4*1483,   1584, 9*1584,   1636, 4*1636,     67, 9*  67,
     4     2027, 4*2027,   2029, 4*2029,    130, 4* 130,   2040, 1*2040,
     5     2049, 1*2049,   2058, 1*2058,   2066, 1*2066,   2076, 1*2076,
     6     2092, 1*2092,   2113, 1*2113,   2135, 1*2135,   2156, 1*2156,
     7     2175, 1*2175,   2194, 1*2194,   2208, 1*2208,   2222, 1*2222,
     8     2239, 1*2239,   2249, 1*2249,    699, 4* 699,    882,    922,
     9      950,    991,   1027,   1061, 4*1061,   1208, 9*1208,   1483/
      END


      BLOCK DATA
C
C         VARIABLE SIZE GRID ID NUMBERS  FOR ST. LOUIS
C         BLOCK DATA PROGRAM NUMBER TWO
      INTEGER A9,A10,A11,A12,A13,A14,A15,A16,A17,A18,A19
      DIMENSION    A9(265),
     2             A10(254), A11(255), A12(229), A13(242), A14(211),
     3             A15(203), A16(202), A17(202), A18(202), A19(212)
      COMMON /VGRID/  ITGRID,   IVG(100,100)
      EQUIVALENCE (A9(1),IVG(82,28))
      EQUIVALENCE (A10(1),IVG(47,31))
      EQUIVALENCE (A11(1),IVG(1,34))
      EQUIVALENCE (A12(1),IVG(56,36))
      EQUIVALENCE (A13(1),IVG(85,38))
      EQUIVALENCE (A14(1),IVG(27,41))
      EQUIVALENCE (A15(1),IVG(38,43))
      EQUIVALENCE (A16(1),IVG(41,45))
      EQUIVALENCE (A17(1),IVG(43,47))
      EQUIVALENCE (A18(1),IVG(45,49))
      EQUIVALENCE (A19(1),IVG(47,51))
C
      DATA A9/
     1   4*1483,   1584, 9*1584,   1636, 4*1636,     67, 9*  67,   2027,
     2   4*2027,   2029, 4*2029,    130, 4* 130,   2040, 1*2040,   2049,
     3   1*2049,   2058, 1*2058,   2066, 1*2066,   2076, 1*2076,   2093,
     4   1*2093,   2114, 1*2114,   2136, 1*2136,   2157, 1*2157,   2176,
     5   1*2176,   2195, 1*2195,   2209, 1*2209,   2223, 1*2223,   2240,
     6   1*2240,   2250, 1*2250,    699, 4* 699,    883, 1* 883,    951,
     7      992,   1028,   1061, 4*1061,   1208, 9*1208,   1483, 4*1483,
8      1584, 9*1584,    1636, 4*1636,      67, 9*  67,    2027, 4*2027,MM  3496
9      2029, 4*2029,     130, 4* 130,    2041, 1*2041,    2050, 1*2050,MM  3497
1      2059, 1*2059,    2067, 1*2067,    2077, 1*2077,    2093, 1*2093,MM  3498
2      2114, 1*2114,    2136, 1*2136,    2157, 1*2157,    2176, 1*2176,MM  3499
3      2195, 1*2195,    2209, 1*2209,    2223, 1*2223,    2240, 1*2240,MM  3500
4      2250, 1*2250,     699, 4* 699,     883, 1* 883,     952,     993,MM  3501
5      1029,    1061, 4*1061,    1208, 9*1208,    1483, 4*1483,    1584,MM  3502
6    9*1584,    1636, 4*1636,      67, 9*  67,      85, 4*  85,    2030,MM  3503
7    4*2030,     131, 4* 131,    2041, 1*2041,    2050, 1*2050,    2059,MM  3504
8    1*2059,    2067, 1*2067,    2077, 1*2077,    2094, 1*2094,    2115,MM  3505
9    1*2115,    2137, 1*2137,    2158, 1*2158,    2177, 1*2177,    2196/MM  3506
  DATA A10/                                                          MM  3507
1    1*2196,    2210, 1*2210,    2224, 1*2224,    2241, 1*2241,    2251,MM  3508
2    1*2251,    2256, 1*2256,    2261, 1*2261,    2262, 1*2262,    2270,MM  3509
3    1*2270,    2279, 1*2279,    1062, 4*1062,    1208, 9*1208,    1484,MM  3510
4    4*1484,    1585, 4*1585,    1619, 4*1619,    1637, 4*1637,      67,MM  3511
5    9*  67,      85, 4*  85,    2030, 4*2030,     131, 4* 131,    2042,MM  3512
6    1*2042,    2051, 1*2051,    2060, 1*2060,    2068, 3*2068,    2094,MM  3513
7    1*2094,    2115, 1*2115,    2137, 1*2137,    2158, 1*2158,    2177,MM  3514
8    1*2177,    2196, 1*2196,    2210, 1*2210,    2224, 1*2224,    2241,MM  3515
9    1*2241,    2251, 1*2251,    2256, 1*2256,    2261, 1*2261,    2262,MM  3516
1    1*2262,    2270, 1*2270,    2279, 1*2279,    1062, 4*1062,    1208,MM  3517
2    9*1208,    1484, 4*1484,    1585, 4*1585,    1619, 4*1619,    1637,MM  3518
3    4*1637,      67, 9*  67,      85, 4*  85,    2030, 4*2030,     131,MM  3519
4    4* 131,    2042, 1*2042,    2051, 1*2051,    2060, 1*2060,    2068,MM  3520
5    3*2068,    2095, 1*2095,    2116, 1*2116,    2138, 1*2138,    2159,MM  3521
6    1*2159,    2178, 1*2178,    2197, 1*2197,    2211, 1*2211,    2225,MM  3522
7    1*2225,    2242, 1*2242,    2252, 1*2252,    2257,    2260, 2*2260,MM  3523
8      2263, 2*2263,    2273, 2*2273,    1062, 4*1062,    1208, 9*1208,MM  3524
9      1484, 4*1484,    1585, 4*1585,    1619, 4*1619,    1637, 4*1637/MM  3525
  DATA A11/                                                          MM  3526
1        67, 9*  67,      85, 4*  85,    2030, 4*2030,     131, 4* 131,MM  3527
2      2043, 1*2043,    2052, 1*2052,    2061, 1*2061,    2068, 3*2068,MM  3528
3      2095, 1*2095,    2116, 1*2116,    2138, 1*2138,    2159, 1*2159,MM  3529
4      2178, 1*2178,    2197, 1*2197,    2211, 1*2211,    2225, 1*2225,MM  3530
5      2242, 1*2242,    2252, 1*2252,    2258,    2260, 2*2260,    2263,MM  3531
6    2*2263,    2273, 2*2273,    1062, 4*1062,    1208, 9*1208,    1484,MM  3532
7    4*1484,    1585, 4*1585,    1619, 4*1619,    1637, 4*1637,      67,MM  3533
8    9*  67,      85, 4*  85,    2030, 4*2030,     131, 4* 131,    2043,MM  3534
9    1*2043,    2052, 1*2052,    2061, 1*2061,    2068, 3*2068,    2096,MM  3535
1    1*2096,    2117, 1*2117,    2139, 1*2139,    2160, 1*2160,    2179,MM  3536
2    1*2179,    2198, 1*2198,    2212, 1*2212,    2226, 1*2226,    2243,MM  3537
3    1*2243,    2253, 1*2253,    2259,    2260, 2*2260,    2263, 2*2263,MM  3538
4      2273, 2*2273,    1062, 4*1062,    1208, 9*1208,    1484, 4*1484,MM  3539
5      1585, 4*1585,    1619, 4*1619,    1637, 4*1637,      68, 4*  68,MM  3540
6        73, 4*  73,      86, 4*  86,     103, 4* 103,     132, 4* 132,MM  3541
7       156, 4* 156,     189, 4* 189,    2096, 1*2096,    2117, 1*2117,MM  3542
8      2139, 1*2139,    2160, 1*2160,    2179, 1*2179,    2198, 1*2198,MM  3543
9      2212, 1*2212,    2226, 1*2226,    2243, 1*2243,    2253, 1*2253/MM  3544
  DATA A12/                                                          MM  3545
1       702,     741,     776,     813, 1* 813,     886, 2* 886,     994,MM  3546
2    1* 994,    1063, 4*1063,    2304, 2*2304,    2313, 1*2313,    1378,MM  3547
3    3*1378,    1459,    1485,    1508,    1517,    1539,    1559,    1586,MM  3548
4    4*1586,    1620, 4*1620,    1637, 4*1637,      68, 4*  68,      73,MM  3549
5    4*  73,      86, 4*  86,     103, 4* 103,     132, 4* 132,     156,MM  3550
```

```
6   4* 156,     189, 4* 189,    2097, 1*2097,    2118, 1*2118,    2140,MM  3551
7   1*2140,    2161, 3*2161,    2199, 1*2199,    2213, 1*2213,    2227,MM  3552
8   1*2227,    2244, 1*2244,    2254, 1*2254,     703,     742,     777,MM  3553
9      813, 1* 813,     886, 2* 886,     994, 1* 994,    1063, 4*1063,MM  3554
1     2304, 2*2304,    2313, 1*2313,    1378, 3*1378,    1460,    1486,MM  3555
2     1509,   1518,    1540,    1560,    1586, 4*1586,    1620, 4*1620,MM  3556
3     1637, 4*1637,      68, 4*  68,      73, 4*  73,      86, 4*  86,MM  3557
4      103, 4* 103,     132, 4* 132,     156, 4* 156,     189, 4* 189,MM  3558
5     2097, 1*2097,    2118, 1*2118,    2140, 1*2140,    2161, 3*2161,MM  3559
6     2199, 1*2199,    2213, 1*2213,    2227, 1*2227,    2244, 1*2244,MM  3560
7     2254, 1*2254,     704,     743,     778,     814,     846,     886,MM  3561
8   2* 886,     995, 1* 995,    1063, 4*1063,    2304, 2*2304,    2314,MM  3562
9     2318,    1378, 3*1378,    1461,    1487,    1510,    1519,    1541/MM  3563
 DATA A13/                                                            MM  3564
1     1561,    1586, 4*1586,    1620, 4*1620,    1637, 4*1637,      68,MM  3565
2   4*  68,      73, 4*  73,      86, 4*  86,     103, 4* 103,     132,MM  3566
3   4* 132,     156, 4* 156,     189, 4* 189,    2098, 1*2098,    2119,MM  3567
4   1*2119,    2141, 1*2141,    2161, 3*2161,    2200, 1*2200,    2214,MM  3568
5   1*2214,    2228, 1*2228,    2245, 1*2245,    2255, 1*2255,     705,MM  3569
6      744,     779,     815,     847,     887, 2* 887,     995, 1* 995,MM  3570
7     1063, 4*1063,    2305, 1*2305,    2310, 1*2310,    2319,    1378,MM  3571
8   3*1378,    1462,    1488,    1511,    1520,    1542,    1562,    1586,MM  3572
9   4*1586,    1620, 4*1620,    1637, 4*1637,      68, 4*  68,      73,MM  3573
1   4*  73,      86, 4*  86,     103, 4* 103,     132, 4* 132,     156,MM  3574
2   4* 156,     189, 4* 189,    2098, 1*2098,    2119, 1*2119,    2141,MM  3575
3   1*2141,    2161, 3*2161,    2200, 1*2200,    2214, 1*2214,    2228,MM  3576
4   1*2228,    2245, 1*2245,    2255, 1*2255,     706,     745,     780,MM  3577
5      816,     848,     887, 2* 887,     996, 1* 996,    1063, 4*1063,MM  3578
6     2305, 1*2305,    2310, 1*2310,    2320,    1379,    1410,    1431,MM  3579
7     1449,    1463,    1489,    1512,    1521,    1543,    1563,    1586,MM  3580
8   4*1586,    1620, 4*1620,    1637, 4*1637,      69, 4*  69,      74,MM  3581
9   4*  74,      87, 4*  87,     104, 4* 104,     133, 4* 133,    2044/MM  3582
 DATA A14/                                                            MM  3583
1   4*2044,     190, 4* 190,     226, 1* 226,     251, 1* 251,     281,MM  3584
2   1* 281,     310, 1* 310,     327,     337,     366,     401,     430,MM  3585
3      472, 1* 472,     540, 1* 540,     597,     635,     665,     707,MM  3586
4      746,     781,     817,     849,     887, 2* 887,     996, 1* 996,MM  3587
5     2284, 1*2284,    2290,    2296, 1*2296,    1211, 1*1211,    1274,MM  3588
6   2*1274,    1380,    1411,    1432,    1450,    1464,    2337, 1*2337,MM  3589
7     2349, 1*2349,    2369, 1*2369,    2388, 3*2388,    1621, 4*1621,MM  3590
8     1638, 4*1638,      69, 4*  69,      74, 4*  74,      87, 4*  87,MM  3591
9      104, 4* 104,     133, 4* 133,    2044, 4*2044,     190, 4* 190,MM  3592
1      226, 1* 226,     251, 1* 251,     281, 1* 281,     310, 1* 310,MM  3593
2      328,     338,     367,     402,     431,     472, 1* 472,     540,MM  3594
3   1* 540,     598,     636,     666,     708,     747,     782,     818,MM  3595
4      850,     888,     923,     953,     997, 1* 997,    2284, 1*2284,MM  3596
5     2291,    2296, 1*2296,    1211, 1*1211,    1274, 2*1274,    1381,MM  3597
6     1412,    1433,    1451,    1465,    2337, 1*2337,    2349, 1*2349,MM  3598
7     2369, 1*2369,    2388, 3*2388,    1621, 4*1621,    1638, 4*1638,MM  3599
8       69, 4*  69,      74, 4*  74,      87, 4*  87,     104, 4* 104,MM  3600
9      133, 4* 133,    2044, 4*2044,     190, 4* 190,     227, 1* 227/MM  3601
 DATA A15/                                                            MM  3602
1      252, 1* 252,     282, 1* 282,     311, 1* 311,     329,     339,MM  3603
2      368,     403,     432,     473,     504,     541,     571,     599,MM  3604
3      637,     667,     709,     748,     783,     819,     851,     889,MM  3605
```

```
     4      924,      954,      997,  1* 997,     2285,     2289,     2292,     2297,MM  3606
     5     2301,     1212,     1246,     1274,   2*1274,     1382,     1413,     1434,MM  3607
     6     1452,     1466,     2338,   1*2338,     2350,   1*2350,     2370,   1*2370,MM  3608
     7     2388,   3*2388,     1621,   4*1621,     1638,   4*1638,       69,  4*   69,MM  3609
     8       74,  4*   74,       87,  4*   87,      104,  4*  104,      133,  4*  133,MM  3610
     9     2044,   4*2044,      190,  4*  190,      227,  1*  227,      252,  1*  252,MM  3611
     1      282,  1*  282,      311,  1*  311,      330,      340,      369,      404,MM  3612
     2      433,      474,      505,      542,      572,      600,      638,      668,MM  3613
     3      710,      749,      784,      820,      852,      890,      925,      955,MM  3614
     4      998,  1*  998,     2286,   1*2286,     2293,     2298,   1*2298,     1213,MM  3615
     5     1247,     1275,     1308,     1340,     1383,     1414,     1435,     1453,MM  3616
     6     1467,     2338,   1*2338,     2350,   1*2350,     2370,   1*2370,     2388,MM  3617
     7   3*2388,     1621,   4*1621,     1638,   4*1638,       69,  4*   69,       74,MM  3618
     8  4*   74,       87,  4*   87,      104,  4*  104,      133,  4*  133,     2044,MM  3619
     9   4*2044,      190,  4*  190,      228,  1*  228,      253,  1*  253,      283/MM  3620
      DATA A16/                                                                     MM  3621
     1  1*  283,      312,  1*  312,      331,      341,      370,      405,      434,MM  3622
     2      475,      506,      543,      573,      601,      639,      669,      711,MM  3623
     3      750,      785,      821,      853,      891,      926,      956,      998,MM  3624
     4  1*  998,     2286,   1*2286,     2294,     2298,   1*2298,     1214,     1248,MM  3625
     5     1276,     1309,     1341,     1384,     1415,     1436,     1454,     1468,MM  3626
     6     2339,   1*2339,     2351,   1*2351,     2371,   1*2371,     2389,   1*2389,MM  3627
     7     2400,   1*2400,     1621,   4*1621,     1638,   4*1638,       60,  4*   60,MM  3628
     8       75,  9*   75,      105,  4*  105,      134,  4*  134,      159,  4*  159,MM  3629
     9      191,  4*  191,      228,  1*  228,      253,  1*  253,      283,  1*  283,MM  3630
     1      312,  1*  312,      332,      342,      371,      406,      435,      476,MM  3631
     2      507,      544,      574,      602,      640,      670,      712,      751,MM  3632
     3      786,      822,      854,      892,      927,      957,      999,     1030,MM  3633
     4     1066,     1095,     1118,     1150,     1179,     1215,     1249,     1277,MM  3634
     5     1310,     1342,     1385,     1416,     1437,     1455,   1*1455,     2339,MM  3635
     6   1*2339,     2351,   1*2351,     2371,   1*2371,     2389,   1*2389,     2400,MM  3636
     7   1*2400,     1622,   4*1622,     1639,   4*1639,       60,  4*   60,       75,MM  3637
     8  9*   75,      105,  4*  105,      134,  4*  134,      159,  4*  159,      191,MM  3638
     9  4*  191,      229,  1*  229,      254,  1*  254,      284,  1*  284,      313/MM  3639
      DATA A17/                                                                     MM  3640
     1  1*  313,      333,      343,      372,      407,      436,      477,      508,MM  3641
     2      545,      575,      603,      641,      671,      713,      752,      787,MM  3642
     3      823,      855,      893,      928,      958,     1000,     1031,     1067,MM  3643
     4     1096,     1119,     1151,     1180,     1216,     1250,     1278,     1311,MM  3644
     5     1343,     1386,     1417,     1438,     1455,   1*1455,     2340,   1*2340,MM  3645
     6     2352,   1*2352,     2372,   1*2372,     2390,   1*2390,     2401,   1*2401,MM  3646
     7     1622,   4*1622,     1639,   4*1639,       60,  4*   60,       75,  9*   75,MM  3647
     8      105,  4*  105,      134,  4*  134,      159,  4*  159,      191,  4*  191,MM  3648
     9      229,  1*  229,      254,  1*  254,      284,  1*  284,      313,  1*  313,MM  3649
     1      334,      344,      373,      408,      437,      478,      509,      546,MM  3650
     2      576,      604,      642,      672,      714,      753,      788,      824,MM  3651
     3      856,      894,      929,      959,     1001,     1032,     1068,     1097,MM  3652
     4     1120,     1152,     1181,     1217,     1251,     1279,     1312,     1344,MM  3653
     5     1387,     1418,     1439,     1456,   1*1456,     2340,   1*2340,     2352,MM  3654
     6   1*2352,     2372,   1*2372,     2390,   1*2390,     2401,   1*2401,     1622,MM  3655
     7   4*1622,     1639,   4*1639,       60,  4*   60,       75,  9*   75,      105,MM  3656
     8  4*  105,      134,  4*  134,      159,  4*  159,      191,  4*  191,      230,MM  3657
     9  1*  230,      255,  1*  255,      285,  1*  285,      314,  1*  314,      335/MM  3658
      DATA A18/                                                                     MM  3659
     1      345,      374,      409,      438,      479,      510,      547,      577,MM  3660
     2      605,     643,     673,     715,     754,     789,     825,     857,MM  3661
     3      895,     930,     960,    1002,    1033,    1069,    1098,    1121,MM  3662
     4     1153,    1182,    1218,    1252,    1280,    1313,    1345,    1388,MM  3663
     5   1*1388,    1440,    1456,  1*1456,    2341,  1*2341,    2353,  1*2353,MM  3664
     6     2373,  1*2373,    2391,  1*2391,    2402,  1*2402,    1622,  4*1622,MM  3665
     7     1639,  4*1639,      60,  4*  60,      75,  9*  75,     105,  4* 105,MM  3666
     8      134,  4* 134,     159,  4* 159,     191,  4* 191,     230,  1* 230,MM  3667
     9      255,  1* 255,     285,  1* 285,     314,  1* 314,     336,     346,MM  3668
     1      375,     410,     439,     480,     511,     548,     578,     606,MM  3669
     2      644,     674,     716,     755,     790,     826,     858,     896,MM  3670
     3      931,     961,    1003,    1034,    1070,    1099,    1122,    1154,MM  3671
     4     1183,    1219,    1253,    1281,    1314,    1346,    1388,  1*1388,MM  3672
     5     1441,    1457,    1469,    2341,  1*2341,    2353,  1*2353,    2373,MM  3673
     6   1*2373,    2391,  1*2391,    2402,  1*2402,    1622,  4*1622,    1639,MM  3674
     7   4*1639,      60,  4*  60,      75,  9*  75,     106,  4* 106,     135,MM  3675
     8   4* 135,     160,  4* 160,     192,  4* 192,    2099,    2109,    2120,MM  3676
     9     2130,    2142,    2150,    2162,    2170,    2180,    2189,     376/MM  3677
      DATA A19/                                                                MM  3678
     1      411,     440,     481,     512,     549,     579,     607,     645,MM  3679
     2      675,     717,     756,     791,     827,     859,     897,     932,MM  3680
     3      962,    1004,    1035,    1071,    1100,    1123,  2*1123,    1220,MM  3681
     4     1254,    1282,    1315,    1347,    2322,    2327,  1*2327,    2331,MM  3682
     5   1*2331,    1494,  2*1494,    1544,    1564,    1590,  4*1590,    1623,MM  3683
     6   4*1623,    1640,  4*1640,      60,  4*  60,      75,  9*  75,     106,MM  3684
     7   4* 106,     135,  4* 135,     160,  4* 160,     192,  4* 192,    2100,MM  3685
     8   1*2100,    2121,  1*2121,    2143,  1*2143,    2163,  1*2163,    2181,MM  3686
     9   1*2181,     377,     412,     441,     482,     513,     550,     580,MM  3687
     1      608,     646,     676,     718,     757,     792,     828,     860,MM  3688
     2      898,     933,     963,    1005,    1036,    1072,    1101,    1123,MM  3689
     3   2*1123,    1221,    1255,    1283,    1316,    1348,    2323,    2327,MM  3690
     4   1*2327,    2331,  1*2331,    1494,  2*1494,    1545,    1565,    1590,MM  3691
     5   4*1590,    1623,  4*1623,    1640,  4*1640,      60,  4*  60,      75,MM  3692
     6   9*  75,     106,  4* 106,     135,  4* 135,     160,  4* 160,     192,MM  3693
     7   4* 192,    2100,  1*2100,    2121,  1*2121,    2143,  1*2143,    2163,MM  3694
     8   1*2163,    2181,  1*2181,     378,     413,     442,     483,     514,MM  3695
     9      551,     581,     609,     647,     677,     719,     758,     793/MM  3696
      END                                                                      MM  3697


      BLOCK DATA                                                               MM  3698
C                                                                              MM  3699
C          BLOCK DATA PROGRAM NUMBER THREE                                     MM  3700
C          VARIABLE SIZE GRID ID NUMBERS FOR ST. LOUIS                         MM  3701
      INTEGER A20,A21,A22,A23,A24,A25,A26,A27,A28,A29,A30                      MM  3702
      DIMENSION   A20(212), A21(206), A22(210), A23(235), A24(230),            MM  3703
     5            A25(220), A26(219), A27(224), A28(266), A29(281),            MM  3704
     6            A30(317)                                                     MM  3705
      COMMON /VGRID/  ITGRID,   IVG(100,100)                                   MM  3706
      EQUIVALENCE (A20(1),IVG(59,53))                                          MM  3707
      EQUIVALENCE (A21(1),IVG(71,55))                                          MM  3708
      EQUIVALENCE (A22(1),IVG(77,57))                                          MM  3709
      EQUIVALENCE (A23(1),IVG(87,59))                                          MM  3710
      EQUIVALENCE (A24(1),IVG(22,62))                                          MM  3711
      EQUIVALENCE (A25(1),IVG(52,64))                                          MM  3712
      EQUIVALENCE (A26(1),IVG(72,66))                                 MM  3713
      EQUIVALENCE (A27(1),IVG(91,68))                                 MM  3714
      EQUIVALENCE (A28(1),IVG(15,71))                                 MM  3715
      EQUIVALENCE (A29(1),IVG(81,73))                                 MM  3716
      EQUIVALENCE (A30(1),IVG(62,76))                                 MM  3717
C                                                                     MM  3718
      DATA A20/                                                       MM  3719
     1    829,    861,    899,    934,    964,   1006,   1037,   1073,MM  3720
     2   1102,   1123, 2*1123,   1222, 2*1222,   1317,   1349,   2324,MM  3721
     3   2328,   2330,   2332,   2334,   1494, 2*1494,   1546,   1566,MM  3722
     4   1590, 4*1590,   1623, 4*1623,   1640, 4*1640,     60, 4*  60,MM  3723
     5     75, 9*  75,    106, 4* 106,    135, 4* 135,    160, 4* 160,MM  3724
     6    192, 4* 192,   2101, 1*2101,   2122, 1*2122,   2144, 1*2144,MM  3725
     7   2164, 1*2164,   2182, 1*2182,    379,    414,    443,    484,MM  3726
     8    515,    552,    582,    610,    648,    678,    720,    759,MM  3727
     9    794,    830,    862,    900,    935,    965,   1007,   1038,MM  3728
     1   1074,   1103,   1124,   1155,   1184,   1222, 2*1222,   1318,MM  3729
     2   1350,   2325,   2329, 1*2329,   2333, 1*2333,   1495,   1513,MM  3730
     3   1522,   1547,   1567,   1590, 4*1590,   1623, 4*1623,   1640,MM  3731
     4 4*1640,     60, 4*  60,     75, 9*  75,    106, 4* 106,    135,MM  3732
     5 4* 135,    160, 4* 160,    192, 4* 192,   2101, 1*2101,   2122,MM  3733
     6 1*2122,   2144, 1*2144,   2164, 1*2164,   2182, 1*2182,    380,MM  3734
     7    415,    444,    485,    516,    553,    583,    611,    649,MM  3735
     8    679,    721,    760,    795,    831,    863,    901,    936,MM  3736
     9    966,   1008,   1039,   1075,   1104,   1125,   1156,   1185/MM  3737
      DATA A21/                                                       MM  3738
     1   1222, 2*1222,   1319,   1351,   2326,   2329, 1*2329,   2333,MM  3739
     2 1*2333,   1496,   1514,   1523,   1548,   1568,   1590, 4*1590,MM  3740
     3   1623, 4*1623,   1640, 4*1640,     70, 4*  70,     76, 9*  76,MM  3741
     4    107, 4* 107,    136, 4* 136,   2045, 2*2045,   2055, 2*2055,MM  3742
     5   2069, 2*2069,   2083,   2102, 1*2102,   2123, 1*2123,   2145,MM  3743
     6 1*2145,   2165, 1*2165,   2183, 1*2183,    381,    416,    445,MM  3744
     7    486,    517,    554,    584,    612,    650,    680,    722,MM  3745
     8    761,    796,    832,    864,    902,    937,    967,   1009,MM  3746
     9   1040,   1076,   1105,   1126,   1157,   1186,   1223,   1256,MM  3747
     1   1284,   1320,   1352,   1391, 4*1391,   1497,   1515,   1524,MM  3748
     2   1549,   1569,   1591, 4*1591,   1624, 4*1624,   1641, 4*1641,MM  3749
     3     70, 4*  70,     76, 9*  76,    107, 4* 107,    136, 4* 136,MM  3750
     4   2045, 2*2045,   2055, 2*2055,   2069, 2*2069,   2084,   2102,MM  3751
     5 1*2102,   2123, 1*2123,   2145, 1*2145,   2165, 1*2165,   2183,MM  3752
     6 1*2183,    382,    417,    446,    487,    518,    555,    585,MM  3753
     7    613,    651,    681,    723,    762,    797,    833,    865,MM  3754
     8    903,    938,    968,   1010, 1*1010,   1077,   1106,   1127,MM  3755
     9   1158,   1187,   1224,   1257,   1285,   1321,   1353,   1391/MM  3756
      DATA A22/                                                       MM  3757
     1 4*1391,   1498,   1516,   1525,   1550,   1570,   1591, 4*1591,MM  3758
     2   1624, 4*1624,   1641, 4*1641,     70, 4*  70,     76, 9*  76,MM  3759
     3    107, 4* 107,    136, 4* 136,   2045, 2*2045,   2055, 2*2055,MM  3760
     4   2069, 2*2069,   2085,   2103, 1*2103,   2124, 1*2124,   2146,MM  3761
     5 1*2146,   2166, 1*2166,   2184, 1*2184,    383,    418,    447,MM  3762
     6    488,    519,    556,    586,    614,    652,    682,    724,MM  3763
     7    763,    798,    834,    866,    904,    939,    969,   1010,MM  3764
     8 1*1010,   1078,   1107,   1128,   1159,   1188,   1225,   1258,MM  3765
     9   1286,   1322,   1354,   1391, 4*1391,   1499, 2*1499,   1551,MM  3766
     1   1571,   1591, 4*1591,   1624, 4*1624,   1641, 4*1641,     70,MM  3767
```

```
2   4*  70,       76,   9*  76,      107,   4* 107,      136,   4* 136,     2046,MM  3768
3   1*2046,     2053,   1*2053,     2062,   1*2062,     2070,   1*2070,     2078,MM  3769
4   1*2078,     2103,   1*2103,     2124,   1*2124,     2146,   1*2146,     2166,MM  3770
5   1*2166,     2184,   1*2184,      384,      419,      448,      489,      520,MM  3771
6      557,      587,      615,      653,      683,      725,      764,      799,MM  3772
7      835,      867,      905,      940,      970,     1010,   1*1011,     1079,MM  3773
8   1*1079,     1129,     1160,     1189,     1226,     1259,     1287,     1323,MM  3774
9     1355,     1391,   4*1391,     1499,   2*1499,     1552,     1572,     1591/MM  3775
 DATA A23/                                                                       MM  3776
1   4*1591,     1624,   4*1624,     1641,   4*1641,       70,   4*  70,       76,MM  3777
2   9*  76,      107,   4* 107,      136,   4* 136,     2046,   1*2046,     2053,MM  3778
3   1*2053,     2062,   1*2062,     2070,   1*2070,     2078,   1*2078,     2104,MM  3779
4   1*2104,     2125,   1*2125,     2147,   1*2147,     2167,   1*2167,     2185,MM  3780
5   1*2185,      385,      420,      449,      490,      521,      558,      588,MM  3781
6      616,      654,      684,      726,      765,      800,      836,      868,MM  3782
7      906,      941,      971,     1011,   1*1011,     1079,   1*1079,     1130,MM  3783
8     1161,     1190,     1227,     1260,     1288,     1324,     1356,     1391,MM  3784
9   4*1391,     1499,   2*1499,     1553,     1573,     1591,   4*1591,     1624,MM  3785
1   4*1624,     1641,   4*1641,       71,   4*  71,       76,   9*  76,      108,MM  3786
2   4* 108,      137,   4* 137,      163,   4* 163,      195,   4* 195,     2104,MM  3787
3   1*2104,     2125,   1*2125,     2147,   1*2147,     2167,   1*2167,     2185,MM  3788
4   1*2185,      386,      421,      450,      491,      522,      559,      589,MM  3789
5      617,      655,      685,      727,      766,      801,      837,      869,MM  3790
6      907,      942,      972,     1012,     1041,     1080,   1*1080,     1131,MM  3791
7     1162,     1191,     1228,     1261,     1289,     1325,     1357,     1392,MM  3792
8   4*1392,     1500,   4*1500,     1592,   4*1592,     1625,   4*1625,     1641,MM  3793
9   4*1641,       71,   4*  71,       76,   9*  76,      108,   4* 108,      137/MM  3794
 DATA A24/                                                                       MM  3795
1   4* 137,      163,   4* 163,      195,   4* 195,     2105,   1*2105,     2126,MM  3796
2   1*2126,     2148,   1*2148,     2168,   1*2168,     2186,   1*2186,      387,MM  3797
3      422,      451,      492,      523,      560,      590,      618,      656,MM  3798
4      686,      728,      767,      802,      838,      870,      908,      943,MM  3799
5      973,     1013,     1042,     1080,   1*1080,     1132,     1163,     1192,MM  3800
6     1229,     1262,     1290,     1326,     1358,     1392,   4*1392,     1500,MM  3801
7   4*1500,     1592,   4*1592,     1625,   4*1625,     1641,   4*1641,       71,MM  3802
8   4*  71,       76,   9*  76,      108,   4* 108,      137,   4* 137,      163,MM  3803
9   4* 163,      195,   4* 195,     2105,   1*2105,     2126,   1*2126,     2148,MM  3804
1   1*2148,     2168,   1*2168,     2186,   1*2186,      388,      423,      452,MM  3805
2      493,      524,      561,      591,      619,      657,      687,      729,MM  3806
3      768,      803,      839,      871,      909,      944,      974,     1014,MM  3807
4     1043,     1081,   2*1081,     1164,     1193,     1230,     1263,     1291,MM  3808
5     1327,     1359,     1392,   4*1392,     1500,   4*1500,     1592,   4*1592,MM  3809
6     1625,   4*1625,     1641,   4*1641,       71,   4*  71,       76,   9*  76,MM  3810
7      108,   4* 108,      137,   4* 137,      163,   4* 163,      195,   4* 195,MM  3811
8     2106,   1*2106,     2127,   1*2127,     2149,   1*2149,     2169,   1*2169,MM  3812
9     2187,   1*2187,      389,   1* 389,      453,      494,      525,      562/MM  3813
 DATA A25/                                                                       MM  3814
1      592,      620,      658,      688,      730,      769,      804,      840,MM  3815
2      872,      910,      945,      975,     1015,     1044,     1081,   2*1081,MM  3816
3     1165,     1194,     1231,     1264,     1292,     1328,     1360,     1392,MM  3817
4   4*1392,     1500,   4*1500,     1592,   4*1592,     1625,   4*1625,     1641,MM  3818
5   4*1641,       71,   4*  71,       76,   9*  76,      108,   4* 108,      137,MM  3819
6   4* 137,      163,   4* 163,      195,   4* 195,     2106,   1*2106,     2127,MM  3820
7   1*2127,     2149,   1*2149,     2169,   1*2169,     2187,   1*2187,      389,MM  3821
8   1* 389,      454,      495,      526,      563,      593,      621,      659,MM  3822
```

```
     9      689,      731,      770,      805,      841,      873,      911,      946,MM  3823
     1      976,     1016,     1045,     1081,   2*1081,     1166,     1195,     1232,MM  3824
     2     1265,     1293,     1329,     1361,     1392,   4*1392,     1500,   4*1500,MM  3825
     3     1592,   4*1592,     1625,   4*1625,     1641,   4*1641,       61,    4* 61,MM  3826
     4       77,    4* 77,       88,    4* 88,      109,   1* 109,      117,   1* 117,MM  3827
     5      122,     2031,     2035,   2*2035,     2037,   2*2037,     2054,   2*2054,MM  3828
     6      196,   4* 196,      237,      245,      256,      277,      286,      308,MM  3829
     7   4* 308,     2201,   1*2201,     2215,   1*2215,     2229,      564,      594,MM  3830
     8      622,      660,      690,      732,      771,      806,      842,      874,MM  3831
     9     2264,   1*2264,     2274,     2280,   1*2280,     1082,   4*1082,     1233/MM  3832
      DATA A26/                                                                       MM  3833
     1   4*1233,     1393,   4*1393,     2342,   1*2342,     2354,   1*2354,     2374,MM  3834
     2   1*2374,     2392,   1*2392,     2403,   1*2403,     1626,   4*1626,     1642,MM  3835
     3   4*1642,       61,    4* 61,       77,    4* 77,       88,    4* 88,      109,MM  3836
     4   1* 109,      117,   1* 117,      123,     2032,     2035,   2*2035,     2037,MM  3837
     5   2*2037,     2054,   2*2054,      196,   4* 196,      238,      246,      257,MM  3838
     6      278,      287,      308,   4* 308,     2201,   1*2201,     2215,   1*2215,MM  3839
     7     2230,      565,      595,      623,      661,      691,      733,      772,MM  3840
     8      807,      843,      875,     2264,   1*2264,     2275,     2280,   1*2280,MM  3841
     9     1082,   4*1082,     1233,   4*1233,     1393,   4*1393,     2342,   1*2342,MM  3842
     1     2354,   1*2354,     2374,   1*2374,     2392,   1*2392,     2403,   1*2403,MM  3843
     2     1626,   4*1626,     1642,   4*1642,       61,    4* 61,       77,    4* 77,MM  3844
     3       88,    4* 88,      110,   2* 110,      118,   1* 118,     2033,     2035,MM  3845
     4   2*2035,     2037,   2*2037,     2054,   2*2054,      196,   4* 196,      239,MM  3846
     5      247,      258,      279,      288,      308,   4* 308,     2202,   2*2202,MM  3847
     6     2216,     2231,      566,      596,      624,      662,      692,      734,MM  3848
     7      773,      808,      844,      876,     2265,     2271,     2276,   2*2276,MM  3849
     8     1082,   4*1082,     1233,   4*1233,     1393,   4*1393,     2343,   1*2343,MM  3850
     9     2355,   1*2355,     2375,     2382,     2393,   1*2393,     2404,   1*2404/MM  3851
      DATA A27/                                                                       MM  3852
     1     1626,   4*1626,     1642,   4*1642,       61,    4* 61,       77,    4* 77,MM  3853
     2       88,    4* 88,      110,   2* 110,      118,   1* 118,     2034,   1*2034,MM  3854
     3     2036,   1*2036,     2038,   1*2038,     2047,   1*2047,     2056,   1*2056,MM  3855
     4      196,   4* 196,      240,   1* 240,      259,      280,   1* 280,      308,MM  3856
     5   4* 308,     2202,   2*2202,     2217,   1*2217,      567,   1* 567,      625,MM  3857
     6      663,      693,      735,      774,      809,      845,   1* 845,     2266,MM  3858
     7   1*2266,     2276,   2*2276,     1082,   4*1082,     1233,   4*1233,     1393,MM  3859
     8   4*1393,     2343,   1*2343,     2355,   1*2355,     2376,     2383,     2393,MM  3860
     9   1*2393,     2404,   1*2404,     1626,   4*1626,     1642,   4*1642,       61,MM  3861
     1    4* 61,       77,    4* 77,       88,    4* 88,      110,   2* 110,      119,MM  3862
     2      124,     2034,   1*2034,     2036,   1*2036,     2038,   1*2038,     2047,MM  3863
     3   1*2047,     2056,   1*2056,      196,   4* 196,      240,   1* 240,      260,MM  3864
     4      280,   1* 280,      308,   4* 308,     2202,   2*2202,     2217,   1*2217,MM  3865
     5      567,   1* 567,      626,      664,      694,      736,      775,      810,MM  3866
     6      845,   1* 845,     2266,   1*2266,     2276,   2*2276,     1082,   4*1082,MM  3867
     7     1233,   4*1233,     1393,   4*1393,     2344,   1*2344,     2356,     2364,MM  3868
     8     2377,     2384,     2394,     2398,     2405,   1*2405,     1626,   4*1626,MM  3869
     9     1642,   4*1642,       61,    4* 61,       78,    4* 78,       89,    3* 89/MM  3870
      DATA A28/                                                                       MM  3871
     1       94,    3* 94,      120,   1* 120,      140,   4* 140,      166,   4* 166,MM  3872
     2      197,   4* 197,      241,   4* 241,      309,   4* 309,      392,   4* 392,MM  3873
     3      568,   4* 568,      737,   4* 737,      914,   4* 914,     2287,   1*2287,MM  3874
     4     2295,   1*2295,     2302,   1*2302,     2308,     1294,     1330,     1362,MM  3875
     5     1394,     1419,   3*1419,     2344,   1*2344,     2357,     2365,     2378,MM  3876
     6     2385,     2395,     2399,     2405,   1*2405,     1627,   4*1627,     1642,MM  3877
     7  4*1642,      61, 4*  61,      78, 4*  78,      89, 3*  89,      94,MM  3878
     8  3*  94,     120, 1* 120,     140, 4* 140,     166, 4* 166,     197,MM  3879
     9  4* 197,     241, 4* 241,     309, 4* 309,     392, 4* 392,     568,MM  3880
     1  4* 568,     737, 4* 737,     914, 4* 914,    2287, 1*2287,    2295,MM  3881
     2  1*2295,    2302, 1*2302,    2309,    1295,    13[illegible],    1363,    1395,MM  3882
     3    1419, 3*1419,    2345, 1*2345,    2358,    23[illegible],    2379,    2386,MM  3883
     4    2396, 1*2396,    2406, 1*2406,    1627, 4*1627,    1642, 4*1642,MM  3884
     5      61, 4*  61,      78. 4*  78,      89, 3*  89,      94, 3*  94,MM  3885
     6     121, 1* 121,     140, 4* 140,     166, 4* 166,     197, 4* 197,MM  3886
     7     241, 4* 241,     309, 4* 309,     392, 4* 392,     568, 4* 568,MM  3887
     8     737, 4* 737,     914, 4* 914,    2288, 2*2288,    2299,    2303,MM  3888
     9    1235,    1266,    1296,    1332,    1364,    1396,    1419, 3*1419/MM  3889
      DATA A29/                                                           MM  3890
     1    2345, 1*2345,    2359,    2367,    2380,    2387,    2396, 1*2396,MM  3891
     2    2406, 1*2406,    1627, 4*1627,    1642, 4*1642,      61, 4*  61,MM  3892
     3      78, 4*  78,      89, 3*  89,      94, 3*  94,     121, 1* 121,MM  3893
     4     140, 4* 140,     166, 4* 166,     197, 4* 197,     241, 4* 241,MM  3894
     5     309, 4* 309,     392, 4* 392,     568, 4* 568,     737, 4* 737,MM  3895
     6     914, 4* 914,    2288, 2*2288,    2300, 1*2300,    1236,    1267,MM  3896
     7    1297,    1333,    1365,    1397,    1419, 3*1419,    2346, 1*2346,MM  3897
     8    2360, 1*2360,    2381, 1*2381,    2397, 1*2397,    2407, 1*2407,MM  3898
     9    1627, 4*1627,    1642, 4*1642,      61, 4*  61,      78, 4*  78,MM  3899
     1      90, 1*  90,      93, 1*  93,      95, 5*  95,     140, 4* 140,MM  3900
     2     166, 4* 166,     197, 4* 197,     241, 4* 241,     309, 4* 309,MM  3901
     3     392, 4* 392,     568, 4* 568,     737, 4* 737,     914, 4* 914,MM  3902
     4    2288, 2*2288,    2300, 1*2300,    1237,    1268,    1298,    1334,MM  3903
     5    1366,    1398,    1420,    1442, 2*1442,    2346, 1*2346,    2360,MM  3904
     6  1*2360,    2381, 1*2381,    2397, 1*2397,    2407, 1*2407,    1627,MM  3905
     7  4*1627,    1642, 4*1642,       0, 9*   0,      90, 1*  90,      93,MM  3906
     8  1*  93,      95, 5*  95,     141, 4* 141,       0, 9*   0,     242,MM  3907
     9  9* 242,     393, 4* 393,     569, 4* 569,     738, 4* 738,     915/MM  3908
      DATA A30/                                                           MM  3909
     1  2* 915,    1017,    1046,    1085,    1108,    1133,    1167,    1196,MM  3910
     2    1238,    1269,    1299,    1335,    1367,    1399,    1421,    1442,MM  3911
     3  2*1442,    1505, 4*1505,    1595, 9*1595,    1643, 4*1643,       0,MM  3912
     4  9*   0,      91, 3*  91,      95, 5*  95,     141, 4* 141,       0,MM  3913
     5  9*   0,     242, 9* 242,     393, 4* 393,     569, 4* 569,     738,MM  3914
     6  4* 738,     915, 2* 915,    1018,    1047,    1086,    1109,    1134,MM  3915
     7    1168,    1197,    1239,    1270,    1300,    1336,    1368,    1400,MM  3916
     8    1422,    1442, 2*1442,    1505, 4*1505,    1595, 9*1595,    1643,MM  3917
     9  4*1643,       0, 9*   0,      91, 3*  91,      95, 5*  95,     141,MM  3918
     1  4* 141,       0, 9*   0,     242, 9* 242,     393, 4* 393,     569,MM  3919
     2  4* 569,     738, 4* 738,     915, 2* 915,    1019,    1048,    1087,MM  3920
     3    1110,    1135,    1169,    1198,    1240,    1271,    1301,    1337,MM  3921
     4    1369,    1401,    1423,    1443,    1458,    1470,    1505, 4*1505,MM  3922
     5    1595, 9*1595,    1643, 4*1643,       0, 9*   0,      91, 3*  91,MM  3923
     6      95, 5*  95,     141, 4* 141,       0, 9*   0,     242, 9* 242,MM  3924
     7     393, 4* 393,     569, 4* 569,     738, 4* 738,     916, 1* 916,MM  3925
     8     977,    1020,    1049,    1088,    1111,    1136,    1170,    1199,MM  3926
     9    1241,    1272,    1302,    1338,    1370,    1402,    1424,    1444/MM  3927
      END                                                                 MM  3928
      BLOCK DATA                                                        MM  3929
C                                                                       MM  3930
C         BLOCK DATA PROGRAM NUMBER FOUR                                MM  3931
C         VARIABLE SIZE GRID ID NUMBERS  FOR ST. LOUIS                  MM  3932
      INTEGER A31, A32, A33                                             MM  3933
      DIMENSION             A31(393), A32(524), A33(1205)               MM  3934
      COMMON /VGRID/  ITGRID,   IVG(100,100)                            MM  3935
      EQUIVALENCE (A31(1),IVG(79,79))                                   MM  3936
      EQUIVALENCE (A32(1),IVG(72,83))                                   MM  3937
      EQUIVALENCE (A33(1),IVG(96,88))                                   MM  3938
C                                                                       MM  3939
      DATA A31/                                                         MM  3940
     1   1*1444,    1471,    1505, 4*1505,    1595, 9*1595,    1643, 4*1643,MM  3941
     2        0, 9*   0,      91, 3*  91,      95, 5*  95,     141, 4* 141,MM  3942
     3        0, 9*   0,     242, 9* 242,     393, 4* 393,     569, 4* 569,MM  3943
     4      738, 4* 738,     916, 1* 916,     978,    1021,    1050,    1089,MM  3944
     5     1112,    1137,    1171,    1200,    1242,    1273,    1303,    1339,MM  3945
     6     1371,    1403,    1425,    1444, 1*1444,    1472,    1505, 4*1505,MM  3946
     7     1595, 9*1595,    1643, 4*1643,       0,34*   0,     242, 9* 242,MM  3947
     8      394, 4* 394,    2234, 4*2234,     739, 4* 739,     917, 1* 917,MM  3948
     9      979,    1022,    1051,    1090,    1113,    1138,    1172,    1201,MM  3949
     1     2306, 1*2306,    2311,    2315,    2321,    1404,    1426,    1445,MM  3950
     2   1*1445,    1473,    1506, 4*1506,    1595, 9*1595,    1643, 4*1643,MM  3951
     3        0,34*   0,     242, 9* 242,     394, 4* 394,    2234, 4*2234,MM  3952
     4      739, 4* 739,     917, 1* 917,     980,    1023,    1052,    1091,MM  3953
     5     1114,    1139,    1173,    1202,    2306, 1*2306,    2312,    2316,MM  3954
     6   1*2316,    1405,    1427,    1445, 1*1445,    1474,    1506, 4*1506,MM  3955
     7     1595, 9*1595,    1643, 4*1643,       0,34*   0,     242, 9* 242,MM  3956
     8      394, 4* 394,    2234, 4*2234,     739, 4* 739,     918, 1* 918,MM  3957
     9      981,    1024,    1053,    1092, 1*1092,    1140, 2*1140,    2307/MM  3958
      DATA A32/                                                         MM  3959
     1   2*2307,    2316, 1*2316,    1406,    1428,    1446, 2*1446,    1506,MM  3960
     2   4*1506,    1595, 9*1595,    1643, 4*1643,       0,34*   0,     242,MM  3961
     3   9* 242,     394, 4* 394,    2234, 4*2234,     739, 4* 739,     918,MM  3962
     4   1* 918,     982,    1025,    1054,    1092, 1*1092,    1140, 2*1140,MM  3963
     5     2307, 2*2307,    2317, 1*2317,    1407,    1429,    1446, 2*1446,MM  3964
     6     1506, 4*1506,    1595, 9*1595,    1643, 4*1643,       0,34*   0,MM  3965
     7      242, 9* 242,     394, 4* 394,    2234, 4*2234,     739, 4* 739,MM  3966
     8     2267, 1*2267,     983,    2281, 1*2281,    1093,    1115,    1140,MM  3967
     9   2*1140,    2307, 2*2307,    2317, 1*2317,    1408,    1430,    1446,MM  3968
     1   2*1446,    1506, 4*1506,    1595, 9*1595,    1643, 4*1643,       0,MM  3969
     2  54*   0,     740, 4* 740,    2267, 1*2267,    2277,    2261, 1*2281,MM  3970
     3     1094, 4*1094,    1245, 4*1245,    1409, 4*1409,    1507, 4*1507,MM  3971
     4     1596, 4*1596,    1628, 4*1628,    1644, 4*1644,       0,54*   0,MM  3972
     5      740, 4* 740,    2268, 2*2268,    2282, 1*2282,    1094, 4*1094,MM  3973
     6     1245, 4*1245,    1409, 4*1409,    1507, 4*1507,    1596, 4*1596,MM  3974
     7     1628, 4*1628,    1644, 4*1644,       0,54*   0,     740, 4* 740,MM  3975
     8     2268, 2*2268,    2282, 1*2282,    1094, 4*1094,    1245, 4*1245,MM  3976
     9     1409, 4*1409,    1507, 4*1507,    1596, 4*1596,    1628, 4*1628/MM  3977
      DATA A33/                                                         MM  3978
     1     1644, 4*1644,       0,54*   0,     740, 4* 740,    2268, 2*2268,MM  3979
     2     2283, 1*2283,    1094, 4*1094,    1245, 4*1245,    1409, 4*1409,MM  3980
     3     1507, 4*1507,    1596, 4*1596,    1628, 4*1628,    1644, 4*1644,MM  3981
     4        0,54*   0,     740, 4* 740,    2269,    2272,    2278,    2283,MM  3982
     5   1*2283,    1094, 4*1094,    1245, 4*1245,    1409, 4*1409,    1507,MM  3983

     6   4*1507,    1596, 4*1596,    1628, 4*1628,    1644, 4*1644,  100*0,MM  3984
     7   100*0, 100*0, 100*0, 100*0, 100*0, 100*0, 100*0, 100*0, 100*0/ MM  3985
      END                                                               MM  3986
```

2. Emission Module Listing

```
      PROGRAM EMMOD
C
C       EMMOD IS THE DRIVER FOR THE   ERT  EMISSIONS MODULE.
C
C       THIS IS A SPECIAL VERSION OF EMMOD SUITABLE ONLY FOR
C       CREATING EMISSION FLUX HISTORIES FROM THE R PS DATA BASE.
C
C       SUBROUTINES REQUIRED
C             ADHOUR      AREAEM      GRIDIT     PONTEM     LOCATE
C             PLUMAS      PARTIT      GESTAB     SEGMET     STACKS
C             PREDAT      NEWPAG      XMIT       MDATE      BLOCKDATA
C             SECOND      DHPLUM
C
C       FUNCTIONS REQUIRED
C             ERF         TRPLOG      DISTAN
C
C       PERIPHERALS REQUIRED
C             TAPE1 = PUNCHED OUTPUT    (UNIT LPUNCH)
C             TAPE5 = INPUT
C             TAPE3 = INTERMEDIATE I/O    (UNIT LIN)
C             TAPE6 = OUTPUT      (UNIT LOUT)
C             LUGRID = LOGICAL UNIT OF GRID FILE
C             LUAREA = LOGICAL UNIT OF AREA  SOURCE EMISSIONS FILE
C             LUPONT = LOGICAL UNIT OF POINT SOURCE EMISSIONS FILE
      COMMON /HISTRY/  TIMEHS(520),       IDHS(520),       UTMHS(2,520),
     1                 NUMHS
      COMMON /TABLES/  THOUR(24)
      COMMON /INPUTS/  TITLE(20),         JDATE(10),       NCASE
      COMMON /LABLIN/  NAMIN(11),         NFLXIN,          WTMOLI(11),
     1                 ADJUST(11)
      COMMON /LABOUT/  NAMOUT(7),         NFXOUT,          WTMOLO(7)
      COMMON /ANSWER/  YES,               RNEG,            IYES,
     1                 NO,                SMALL
      COMMON /WIND/    T(100),            V(100),          TH(100),
     1                 NPTS
      COMMON /TRAJ/    P(2,100)
      COMMON /PARCEL/  PWIDTH,            HPWIDT,          PLENTH,
     1                 PAREA,             DTFREZ,         SIGEOG,
     2                 ZEE(5),            HTINTF(6),       NOSTAT,
     3                 ROAIR
      COMMON /TEMPS/   TMTEMP(25),        TEMPSF(25),      NTEMP,
     1                 ITEMP
      COMMON /MXHITE/  TMXHIT(25),        HTMIXL(25),      NMIXL,
     1                 IMIXL
      COMMON /ORIGIN/  UTMXOR,            UTMYOR
      COMMON /PASQAL/  TMSTAB(25),        KSTABL(25),      ISTBMX
      COMMON /EMRATE/  EMARAT(7,520),     EMABAS(7),       NOSUMS
C
      DATA YES, RNEG, IYES, NO /4HYES ,4HNO  ,4HYES ,4HNO   /
      DATA RMOR, TEND   /4HMORE, 4HEND  /
      DATA NCASE, KOK /1,-1/
      DATA THOUR /0.0,60.0,120.0,180.0,240.0,300.0,360.0,420.0,480.0,
     1 540.0,600.0,660.0,720.0,780.0,840.0,900.0,960.0,1020.0,1080.0,
     1 1140.0,1200.0,1260.0,1320.0,1380.0/
      DATA  SMALL,  ROAIR,  RAD  /1.E-4,  1178.,  .01745322925 /
      DATA SIGEDG, DTFREZ /2.15 , 60./
      DATA LIN, LOUT /3,6/
      TERM = TEND
      DEBUG =YES
      CALL MDATE(JDATE)
C
C        READ GENERAL INPUTS
C
100   CONTINUE
      CALL SECOND(A1)
      CALL FREDAT
      READ(LIN,1)  TITLE
      READ(LIN,2)  NDAREA
      READ(LIN,2)  NOPONT
      READ(LIN,2)  NOSUMS
      READ(LIN,2)  IPUNCH
      READ(LIN,3)  LUGRID
      READ(LIN,3)  LUAREA
      READ(LIN,3)  LUPONT
      READ(LIN,3)  NFLXIN
      READ(LIN,3)  NFXOUT
      READ(LIN,11)  (NAMIN(I),ADJUST(I),I=1,NFLXIN)
      READ(LIN,2)  (NAMOUT(I),I=1,NFXOUT)
      READ(LIN,23)  PLENTH
      READ(LIN,23)  PWIDTH
      PAREA = PLENTH*PWIDTH
      HPWIDT = .50*PWIDTH
      IF(NDAREA.NE.IYES)  GO TO 180
C
C         AREA SOURCE INPUTS AND DETERMINATION
C
      READ(LIN,13)  (EMABAS(I),I=1,NFXOUT)
      DO 120 K = 1,501
      READ(LIN,4) A, B, C, ID
      IF(A.LT.0.)  GO TO 140
      TIMEHS(K) = A
      UTMHS(1,K) = B
      UTMHS(2,K) = C
      IDHS(K) = ID
120   CONTINUE
      WRITE(LOUT,5)
      GO TO 500
140   NUMHS = K-1
C
      CALL ADHOUR
C
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,14)
      DO 160 K = 1,NUMHS
      IF(MOD(K,25).NE.0) GO TO 150
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,14)
150   WRITE(LOUT,6) TIMEHS(K),UTMHS(1,K),UTMHS(2,K),IDHS(K)
160   CONTINUE
      IF(NCASE.GT.1)  GO TO 170
```

```
C
      CALL GRIDIT(LUGRID,KOK)
C
      IF(KOK.LE.0) GO TO 170
      WRITE(LOUT,12)  KOK
      GO TO 500
170   CONTINUE
      IF(NDAREA.NE.IYES) GO TO 180
C
      CALL AREAEM(LUAREA,IPUNCH,KOK)
C
      IF(KOK.LE.0)  GO TO 180
      WRITE(LOUT,8)  KOK
      GO TO 500
180   CONTINUE
      IF(NOPONT.NE.IYES)  GO TO 400
C
C          POINT SOURCE INPUTS AND DETERMINATION
C
      READ(LIN,20)  UTMXOR, UTMYOR
      READ(LIN,20)  P(1,1),  P(2,1)
      DO 200 I = 1,101
      READ(LIN,20)  A, B, C
      IF(A .LT. 0.0)  GO TO 210
      T(I) = A
      V(I) = B/60.
      TH(I) = C*RAD
200   CONTINUE
      WRITE(LOUT,21)
      GO TO 500
210   NPTS = I - 1
      DO 220 I = 2,NPTS
      J = I - 1
      DT = T(I) - T(J)
      DX = V(J)*COS(TH(J))*DT
      DY = V(J)*SIN(TH(J))*DT
      P(1,I) = P(1,J) + DX
      P(2,I) = P(2,J) + DY
220   CONTINUE
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,27)
      DO 230 I = 1,NPTS
      IF(MOD(I,25).NE.0)  GO TO 225
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LUUT,27)
225   VPHR =V(I)*60.
      THDEG = TH(I)/RAD
      WRITE(LOUT,28) I, T(I), P(1,I), P(2,I), VPHR, THDEG
230   CONTINUE
C
      READ(LIN,3)  NOSTAT
      IF(NOSTAT .LE. 5)  GO TO 240
      WRITE(LOUT,22)
      GO TO 500
240   READ(LIN,23)  (ZEE(I),I=1,NOSTAT)
```

```
      HTINTF(1) = ZEE(1)
      DO 250 J = 2,NOSTAT
      I = J - 1
      HTINTF(J) = 0.50*(ZEE(I) + ZEE(J))
  250 CONTINUE
      HTINTF(NOSTAT+1) = ZEE(NOSTAT)
C
C     READ(LIN,23)  DTFREZ
C     READ(LIN,23)  SIGEDG
      DO 260 I = 1,26
      READ(LIN,20)  A,B
      IF(A .LT. -9.)  GO TO 270
      TMTEMP(I) = A
      TEMPSF(I) = B
  260 CONTINUE
      WRITE(LOUT,24)
      GO TO 500
  270 NTEMP = I - 1
      ITEMP = 1
C
      K=1
      DO 280 I = 1,26
      READ(LIN,25) A, IS
      IF(A .LT. 0.)  GO TO 300
      IF(I.EQ.1) GO TO 275
      IF(IS.EQ.KSTABL(K-1))  GO TO 280
  275 CONTINUE
      TMSTAB(K) = A
      KSTABL(K) = IS
           K  = K+1
  280 CONTINUE
      WRITE(LOUT,26)
      GO TO 500
  300 CONTINUE
      ISTBMX= K -1
      DO 310 I = 1,26
      READ(LIN,20) A, B
      IF(A.LT.0.0)  GO TO 320
      TMXHIT(I) = A
      HTMIXL(I) = B
  310 CONTINUE
      WRITE(LOUT,32)
      GO TO 500
  320 NMIXL = I - 1
      IMIXL = 1
C
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,29) (TMSTAB(I), KSTABL(I), I=1,ISTBMX)
      WRITE(LOUT,30) (TMTEMP(I), TEMPSF(I), I=1,NTEMP)
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,31) (TMXHIT(I),HTMIXL(I),I=1,NMIXL)
C
      CALL PONTEM(LUPONT,IPUNCH,KOK,NDSUMS)
C
      IF(KOK.GT.0)  GO TO 400
      WRITE(LOUT,9)  KOK
      GO TO 500
400   CONTINUE
C
      CALL NEWPAG(TITLE,0,JDATE)
      CALL SECOND(A2)
      A2 = A2 - A1
      WRITE(LOUT,10) NCASE, A2
410   READ(LIN,1) TERM
      IF(TERM.EQ.RMOR)  GO TO 420
      IF(TERM.NE.TEND)  GO TO 410
      GO TO 500
420   CONTINUE
      NCASE = NCASE + 1
      GO TO 100
500   STOP
C
1     FORMAT (20A4)
2     FORMAT (40X,A4)
3     FORMAT (40X,I2)
4     FORMAT (10X,3F10.2,I10)
5     FORMAT (54H1TOO MANY GRID SQUARE CROSSINGS INPUT -- JOB ABORTED  )
6     FORMAT (1H0,6X,3F12.2,I12)
7     FORMAT (40X,A4,6X,2F10.0)
8     FORMAT (50H1CALCULATIONS STOPPED IN 'AREAEM'  --  JOB ABORTED ,I9)
9     FORMAT (50H1CALCULATIONS STOPPED IN 'PONTEM'  --  JOB ABORTED ,I9)
10    FORMAT (20H0END OF CASE NUMBER   ,I3,10X,10HCP TIME =  ,F8.2,
     1  5H  SEC )
11    FORMAT (40X,A4,6X,F10.0)
12    FORMAT (50H1CALCULATIONS STOPPED IN 'GRIDIT'  --  JOB ABORTED ,I9)
13    FORMAT (40X,F10.0)
14    FORMAT (1H0,31HTRAJECTORY DATA BY GRID SQUARE     ,
     1  //    ,12X,4HTIME,8X,6HUTM-X ,6X,6HUTM-Y  ,10X,2HID )
20    FORMAT (40X,3F10.0)
21    FORMAT (1H0,48HTOO MANY TRAJECTORY NODES INPUT --  JOB ABORTED  )
22    FORMAT (1H0,48HTOO MANY VERTICAL STATIONS INPUT - JOB ABORTED   )
23    FORMAT (40X,F10.0)
24    FORMAT (1H0,48HTOO MANY TEMPERATURES INPUT  --  JOB ABORTED     )
25    FORMAT (40X,F10.0,I10)
26    FORMAT (1H0,48HTOO MANY STABILITY UPDATES INPUT - JOB ABORTED   )
27    FORMAT(1H08X16HTRAJECTORY DATA  ,//,4X,3HLEG,5X,4HTIME,8X,1HX,10X,
     11HY,10X,8HVELOCITY,6X,5HTHETA,/,4X,3HNO.,5X,5H(MIN),6X,4H(KM),
     2  7X,4H(KM),8X,8H(KM/HR) ,4X,  9H(DEGREES)  )
28    FORMAT (1H0,I5,F11.1,2F11.2,2F13.3)
29    FORMAT(1H0,8X,20HSTABILITY CLASS DATA  ,//,11X,4HTIME,5X,5HCLASS,
     1  25(//,10X,F5.0,7X,I1))
30    FORMAT(1H0,8X,26HSURFACE TEMPERATURE DATA    ,//,11X,4HTIME,7X,
     1  11HTEMPERATURE ,25(//,10X,F5.0,7X,F7.1))
31    FORMAT(1H0,8X,25HMIXING LAYER DEPTH DATA    ,//,11X,4HTIME,6X,
     1  6HDEPTH ,25(//,10X,F5.0,5X,F6.1))
32    FORMAT(51H0TOO MANY MIXING LAYER HEIGHTS INPUT -- JOB ABORTED )
      END
      SUBROUTINE ADHOUR
C
C          *ADHOUR*  INSERTS TRAJECTORY TIMES ON THE EVEN HOURS
C
      DIMENSION  IWORK(520)
      COMMON /HISTRY/ TIMEHS(520),    IDHS(520),      UTMHS(2,520),
     1                NUMHS
      COMMON /TABLES/ THOUR(24)
      COMMON /WORKER/ IDATE,          NG, IHR,        WORK(22440)
      EQUIVALENCE (IWORK,WORK)
C
      K = 1
      JT = IFIX(TIMEHS(1)+SMALL)/60 + 2
      DO 100 I = 1,NUMHS
      IM1 = I - 1
60    IF(TIMEHS(I).LE.THOUR(JT)) GO TO 90
      IWORK(K) = IDHS(IM1)
      WORK(K+520) = THOUR(JT)
      TPORT = (THOUR(JT)-TIMEHS(IM1))/(TIMEHS(I)-TIMEHS(IM1))
      WORK(K+1040) = UTMHS(1,IM1)+TPORT*(UTMHS(1,I)-UTMHS(1,IM1))
      WORK(K+1560) = UTMHS(2,IM1)+TPORT*(UTMHS(2,I)-UTMHS(2,IM1))
      K = K + 1
      JT = JT + 1
      GO TO 60
90    CONTINUE
      IWORK(K) = IDHS(I)
      WORK(K+520) = TIMEHS(I)
      WORK(K+1040) = UTMHS(1,I)
      WORK(K+1560) = UTMHS(2,I)
      K = K + 1
100   CONTINUE
      NUMHS = K - 1
      DO 120 I = 1,NUMHS
      IDHS(I) = IWORK(I)
      TIMEHS(I) = WORK(I+520)
      UTMHS(1,I) = WORK(I+1040)
      UTMHS(2,I) = WORK(I+1560)
120   CONTINUE
      K = NUMHS + 1560
      CALL XMIT (-K,0.,WORK)
      RETURN
      END


      SUBROUTINE AREAEM(LUAREA,IPUNCH,KOK)
C
C        AREAEM READS THE RAPS AREA SOURCE EMISSIONS TAPE AND
C        DETERMINES THE AREA SOURCE EMISSION RATE HISTORY FOR
C        A SPECIFIED TRAJECTORY.
C
C       RAP FILE DESCRIPTION
C
C          IDATE NGRID IHOUR  PART SOX  NOX  THC  CO   DUST
C                             NRHC PARF OLEF AROM ALDE
```

```
C                                PART SOX  .....
C                                                  .....ALDE
C                                (REPEATED NGRID TIMES PER HOUR)
C
      DIMENSION         ASOURC(11,2000),  IDLAST(44),          SUM(510)
     1                , ISUM(3)
      COMMON /DEGRID/ ID(2000),           AREA(2000),        NGRID
      COMMON /INPUTS/ TITLE(20),          JDATE(10),           NCASE
      COMMON /HISTRY/ TIMEHS(520),        IDHS(520),           UTMHS(2,520),
     1                NUMHS
      COMMON /TABLES/ THOUR(24)
      COMMON /PARCEL/ PWIDTH,             HPWIDT,            PLENTH,
     1                PAREA,              DTFREZ,            SIGEDG,
     2                ZEE(5),             HTINTF(6),         NOSTAT,
     3                ROAIR
      COMMON /WORKER/ IDATE,              NG,IHOUR,            WORK(22440)
      COMMON /EMRATE/ EMARAT(7,520),      EMABAS(7),           NDSUMS
      COMMON /ANSWER/ YES,RNEG,           IYES,NO,             SMALL
      COMMON /LABLIN/ NAMIN(11),          NFLXIN,            WTMOLI(11),
     1                ADJUST(11)
      COMMON /LABOUT/ NAMOUT(7),          NFXOUT,            WTMOLO(7)
      EQUIVALENCE   (ASOURC, WORK )
      EQUIVALENCE   (ISUM, SUM)
      DATA ALL, LOUT, LPUNCH /4HALL ,6,1/
      DATA NUMREC, NWPREC /44 510/
      DEBUG = RNEG
      PARERR = RNEG
      NM1 = NUMREC - 1
      IF(NCASE.GT.1)   GO TO
C        GENERATE IDLAST ARR    WHICH CONTAINS THE GRID SQUARE ID NUMBER
C        ASSOCIATED WITH THE LAST SET OF NFLXIN VARIABLE IN EACH RECORD.
      DO 70 J = 1,NUMREC
      JL = (NWPREC*J - 3)/NFLXIN
      IF(JL.LE.NGRID)  GO TO 65
      IDLAST(J) = ID(NGRID) + 1
      GO TO 70
65    CONTINUE
      IDLAST(J) = ID(JL)
70    CONTINUE
      IF(DEBUG.EQ.YES)  WRITE(LOUT,9) (IDLAST(J),J=1,NUMREC)
80    CONTINUE
C
C        SKIP 'PRE-TRAJECTORY START TIME DATA' IN FILE
C
      INDEX = IFIX(TIMEHS(1)+SMALL)/60
      IF(INDEX.EQ.0)  GO TO 125
      IF(INDEX.GT.23) STOP
      DO 120 I = 1,INDEX
CUNIVAC  READ(LUAREA,ERR=100,END=510) (ISUM(JJ),JJ=1,3)
      BUFFER IN (LUAREA,1) (ISUM(1),ISUM(3))
90    IF(UNIT(LUAREA))    95,  510,  100
95    IF(DEBUG.EQ.YES)  WRITE(LOUT,1) (ISUM(JJ),JJ=1,3)
100   DO 110 J = 2,NUMREC
      READ(LUAREA)
110   CONTINUE
```

```
120   CONTINUE
C
125   INEXT = 1
      K2MAX = NFLXIN*2000
      IF(NDSUMS.NE.IYES)  GO TO 130
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,6)  (NAMIN(JJ),JJ=1,NFLXIN)
C
C         HOURLY TIME CYCLE BEGINS HERE
C
130   IHIST = INEXT
      DONTRD = RNEG
      INDEX = INDEX + 1
      IDMAX = -1
      DO 150 I = IHIST,NUMHS
      JT = IFIX(TIMEHS(I)+SMALL)/60 + 1
      IF(JT.LE.INDEX)  GO TO 140
      INEXT = I
      GO TO 160
140   IDMAX = MAX0(IDMAX,IDHS(I))
150   CONTINUE
      INEXT = I + 1
160   IF(IDMAX.GT.0)   GO TO 200
C
C        ALL TRAJECTORY NODES THIS HOUR ARE OFF GRID AND
C        THE DEFAULT AREA SOURCE EMISSION RATES ARE USED
C
      DONTRD = YES
      READ(LUAREA) IDATE , NG , IHOUR
      IF(IOCHEC(LUAREA))  185,180
180   IF(DEBUG.EQ.YES) WRITE(LOUT,1) IDATE,NG,IHOUR
185   CONTINUE
      DO 190 J = 2,NUMREC
      READ(LUAREA)
      IF(IOCHEC(LUAREA))  190,190 .
190   CONTINUE
      GO TO 300
200   CONTINUE
C
C        FIND LAST RECORD TO BE READ FOR THIS HOUR
C
      DO 220 J = 1,NUMREC
      IF(IDMAX.GE.IDLAST(J))  GO TO 220
      LSTREC = J
      GO TO 230
220   CONTINUE
      LSTREC = J
230   CONTINUE
      IF(NDSUMS.EQ.IYES)  LSTREC = NUMREC
      IF(DEBUG.EQ.YES)  WRITE(LOUT,2)  LSTREC
      K2 = NWPREC -3
C
C       ON THE UNIVAC REPLACE BUFFER IN AND IF(UNIT) WITH
C     READ(LUAREA,ERR=245,END=510) IDATE,NG,IHOUR,(SUM(JJ),JJ=1,K2)
C
```

```
      BUFFER IN (LUAREA,1)  (SUM(1),SUM(NWPREC))
240   IF(UNIT(LUAREA))   242,  510,  245
242   IDATE = ISUM(1)
      NG = ISUM(2)
      IHOUR = ISUM(3)
      CALL XMIT(K2,SUM(4),WORK)
      GO TO 250
245   PARERR = YES
      WRITE(LOUT,7)
250   CONTINUE
      IF(LSTREC.EQ.1)  GO TO 275
      DO 270 J = 2,NUMREC
      K1 = K2 + 1
      K2 = K2 + NWPREC
      IF(J.GT.LSTREC)    GO TO 265
C
C        ON THE UNIVAC REPLACE BUFFER IN AND IF(UNIT) WITH
C     READ(LUAREA,ERR=263,END=510) (SUM(JJ),JJ=1,NWPREC)
C
      BUFFER IN (LUAREA,1) (SUM(1), SUM(NWPREC))
260   IF(UNIT(LUAREA))   262,  510,  263
262   CALL XMIT(NWPREC,SUM,WORK(K1))
      GO TO 270
263   PARERR = YES
      WRITE(LOUT,7)
      GO TO 270
265   READ (LUAREA)
      IF(IOCHEC(LUAREA)) 270,270
270   CONTINUE
275   CONTINUE
      IF(DEBUG.NE.ALL)  GO TO 280
      WRITE(LOUT,1) IDATE, NG, IHOUR
      N = LSTREC*NWPREC
      WRITE(LOUT,5) (WORK(K),K=1,N)
280   CONTINUE
      IF(NDSUMS.NE.IYES)  GO TO 300
      IF(PARERR.EQ.YES)  GO TO 300
      DO 290 I = 1,NFLXIN
      SUM(I) = 0.0
      DO 285 J = 1,NGRID
      SUM(I) = SUM(I) + ASOURC(I,J)
285   CONTINUE
      SUM(I) = SUM(I)*ADJUST(I)
290   CONTINUE
      WRITE(LOUT,18) THOUR(INDEX), (SUM(I),I=1,NFLXIN)
C
300   CONTINUE
C
      INM1 = INEXT - 1
      DO 360 I = IHIST,INM1
      IDSQ = IDHS(I)
      IF(IDSQ.LE.0)  GO TO 340
      DO 310 J = 1,NGRID
      IF(IDSQ.NE.ID(J))  GO TO 310
      JDEX = J
      GO TO 320
310   CONTINUE
      GO TO 530
320   CONTINUE
      AREAFR = PAREA/(AREA(JDEX))
C
C     NOX, PARF, OLEF, AROM, ALDE, CO, SOX
C
      EMARAT(1,I) = ASOURC( 3,JDEX)*ADJUST( 3)*AREAFR
      EMARAT(2,I) = ASOURC( 8,JDEX)*ADJUST( 8)*AREAFR
      EMARAT(3,I) = ASOURC( 9,JDEX)*ADJUST( 9)*AREAFR
      EMARAT(4,I) = ASOURC(10,JDEX)*ADJUST(10)*AREAFR
      EMARAT(5,I) = ASOURC(11,JDEX)*ADJUST(11)*AREAFR
      EMARAT(6,I) = ASOURC( 5,JDEX)*ADJUST( 5)*AREAFR
      EMARAT(7,I) = ASOURC( 2,JDEX)*ADJUST( 2)*AREAFR
C
      IF(DEBUG.NE.YES)  GO TO 360
      WRITE(LOUT,4) TIMEHS(I),IDSQ,(EMARAT(J,I),J=1,NFXOUT)
      GO TO 360
340   CONTINUE
      DO 350 J = 1,NFXOUT
      EMARAT(J,I) = EMABAS(J)
350   CONTINUE
      IF(DEBUG.NE.YES)  GO TO 360
      WRITE(LOUT,4) TIMEHS(I),IDSQ,(EMARAT(J,I),J=1,NFXOUT)
360   CONTINUE
      IF(INM1 .LT. NUMHS)   GO TO 130
C
C          OUTPUT
C
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,16) (NAMOUT(K),K=1,NFXOUT)
      DO 380 J =1,NUMHS
      IF(MOD(J,25).NE.0)  GO TO 370
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,16) (NAMOUT(K),K=1,NFXOUT)
370   WRITE(LOUT,17) TIMEHS(J),IDHS(J),(EMARAT(I,J),I=1,NFXOUT)
380   CONTINUE
      CALL XMIT(-NFXOUT,0.0,SUM)
      TIMEHS(NUMHS+1) = TIMEHS(NUMHS) + SMALL
C       CONVERT FROM MOLES/HR TO MOLE FRACTION - METERS/MIN
      A = 28.97/(ROAIR*PAREA*60.E6)
      DO 400 I = 1,NUMHS
      DT = TIMEHS(I+1) - TIMEHS(I)
      DT = DT/60.
      DO 400 J = 1,NFXOUT
      SUM(J) = SUM(J) + EMARAT(J,I)*DT
      EMARAT(J,I) = A*EMARAT(J,I)
400   CONTINUE
      WRITE(LOUT,15) (SUM(K),K=1,NFXOUT)
C
      A = -10.
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,11) (NAMOUT(K),K=1,NFXOUT)
      IF(IPUNCH.EQ.IYES) WRITE(LPUNCH,12)
      DO 420 J = 1,NUMHS                                                EM   545
      IF(MOD(J,25).NE.0)  GO TO 410                                     EM   546
      CALL NEWPAG(TITLE,0,JDATE)                                        EM   547
      WRITE(LOUT,11) (NAMOUT(K),K=1,NFXOUT)                             EM   548
410   WRITE(LOUT,13) TIMEHS(J), (UTMHS(K,J),K=1,2), (EMARAT(K,J),       EM   549
     1                                          K =1,NFXOUT )           EM   550
      IF(IPUNCH.EQ.IYES)  WRITE(LPUNCH,14)                              EM   551
     1                      J,TIMEHS(J),(EMARAT(K,J),K=1,NFXOUT)        EM   552
420   CONTINUE                                                          EM   553
      IF(IPUNCH.EQ.IYES)   WRITE(LPUNCH,14) J,A                         EM   554
      RETURN                                                            EM   555
C                                                                       EM   556
510   KOK = 999                                                         EM   557
      WRITE(LOUT,8)  KOK                                                EM   558
      RETURN                                                            EM   559
520   KOK = 9999                                                        EM   560
      WRITE(LOUT,7)  KOK                                                EM   561
      RETURN                                                            EM   562
530   KOK = 99999                                                       EM   563
      WRITE(LOUT,10)  IOSQ, KOK                                         EM   564
      RETURN                                                            EM   565
C                                                                       EM   566
1     FORMAT (1H0,7HIDATE =,I8,10H   NGRID =  ,I7,10H   IHOUR =   ,I4)  EM   567
2     FORMAT (1H0,18H LAST RECORD NO.=   I4)                            EM   568
4     FORMAT (1H ,6HTIME =,F8.2,6H  ID = ,I6,3X,7E11.3,9H MOLES/HR )    EM   569
5     FORMAT (1H ,11E11.3)                                              EM   570
6     FORMAT (1H0,35HREGIONAL AREA SOURCE EMISSION SUMS ,30X,           EM   571
     1  12H(MOLES/HOUR),//,8H   TIME ,6X,11(A4,7X))                     EM   572
7     FORMAT (50H0 SUBROUTINE AREAEM ENCOUNTERED PARITY ERRORS  --      EM   573
     1 45HSOME DATA FROM PREVIOUS HOUR USED                   I4)       EM   574
8     FORMAT (80H1 SUBROUTINE AREAEM ENCOUNTERED UNEXPECTED EOF  --     EM   575
     1    JOB ABORTED                                   I8)             EM   576
9     FORMAT(1H0,8HIDLAST    )                                          EM   577
10    FORMAT (1H0,//,23H  GRID SQUARE ID NUMBER  ,I6,2X,                EM   578
     1  36HIS NOT ON THE GRID  -- JOB ABORTED    ,/, 6H KOK=  ,I8)      EM   579
11    FORMAT (1H0,27HAREA SOURCE EMISSION FLUXES ,30X,13HMOLE FRACTION  EM   580
     1 17H - METERS/MINUTE   ,//,6X,5HTIME ,5X,5HUTM-X,5X,5HUTM-Y,      EM   581
     2  7(5X,A4,3X))                                                    EM   582
12    FORMAT (30H   AREA SOURCE EMISSIONS                    )          EM   583
13    FORMAT (1H0,3F10.1,7E12.4)                                        EM   584
14    FORMAT (12HAREA SOURCES,I4,4X,F10.2,4E12.4,/,30X,4E12.4)          EM   585
15    FORMAT(1H ,20X,7(2X,10H----------),//,4X,12HTOTAL MOLES   ,       EM   586
     1    5X,7E12.4)                                                    EM   587
16    FORMAT (1H0,46HAREA SOURCE EMISSION RATES    --    MOLES/HR   ,   EM   588
     1  //,5X,5HTIME ,5X,2HID,4X,7(5X,A4,3X))                           EM   589
17    FORMAT (1H0,F10.2,3X,I4,3X,7E12.4)                                EM   590
18    FORMAT (1H0,F7.1,2X,11E11.3)                                      EM   591
      END                                                               EM   592
      SUBROUTINE DHPLUM(TA,TS,U,ISTAB,ACMM,HTINV,HTPS,DH,PENTFR,DTDZ)
C
C        DHPLUM CALCULATES PLUME RISE AND INVERSION PENETRATION.
C
C     INPUTS
C          TA      =  AMBIENT TEMPERATURE  (DEG C)
C          TS      =  STACK GAS TEMPERATURE  (DEG C)
C          U       =  WIND SPEED   (M/SEC)
C          ISTAB   =  ATMOSPHERIC STABILITY CLASS (1-6  FOR  A-F)
C          ACMM    =  STACK VOLUMETRIC FLOW RATE   (M**3/MIN)
C          HTINV   =  INVERSION HEIGHT (METERS)
C          HTPS    =  STACK HEIGHT  (METERS)
C          DTDZ    =  POTENTIAL TEMPERATURE GRADIENT  (DEG C/METER)
C
C     OUTPUTS
C          DH      =  PLUME RISE  ( METERS )
C          PENTFR =  INVERSION PENETRATION FRACTION  (0 TO 1)
C
      DATA ERLMT, NITRAT, YES /.005, 150, 4HYES /
      DATA DFDTDZ, GRAV, BETPSQ /.0137,  9.80665,  .160/
      DATA UMIN /1./
      DEBUG = GRAV
      TAK = TA + 273.2
      TSK = TS + 273.2
      ACMS = ACMM/188.49556
      F = GRAV*ACMS*(TSK-TAK)/TSK
      UU = AMAX1(U,UMIN)
      UDTDZ = DTDZ
      IF(DTDZ.LT.0.0  .OR.  DTDZ.GT.0.10)  UDTDZ = DFDTDZ
      S = GRAV*UDTDZ/TAK
      ZB = HTINV - HTPS
      ZBM = ZB - 5.
      PENTFR = 0.0
      IF(ZBM.LE.0.0)  PENTFR = 1.0000
      IF(UU.GT.1.39   .OR. ISTAB.LE.4)  GO TO 10
C     -------  CALM CONDITIONS  --------
      DH = 5.0*(F/S**1.5)**.25
      GO TO 100
10    IF(ISTAB.LE.4)  GO TO 20
C     -------  STABLE WITH WIND  --------
      DH = 2.9*(F/(UU*S))**.333333
      GO TO 100
20    CONTINUE
C     --------  NEUTRAL OR UNSTABLE WITH WIND, WITHOUT INVERSION -----
C                 (IE....STACK HEIGHT ABOVE INVERSION HEIGHT)
      IF(F.GT.55.) GO TO 30
      XSTAR = 14.*F**.625
      GO TO 40
30    XSTAR = 34.5*F**.40
40    XRSTAR = 3.5*XSTAR
      DH = 1.6*((F*XRSTAR**2)**.333333)/UU
      IF(ZBM.LE.0.0)   GO TO 100
      DHMAX = DH
C     -------  NEUTRAL OR UNSTABLE WITH WIND BELOW A STABLE LAYER  -----
C              DEFINE ZR = ZEQ/ZB
```

```
C                A  = (3*F/(BETPSQ*U*S*ZB**3)-2/9)*.666666            EM   648
C           SOLVE FOR THE ROOTS OF                                     EM   649
C                FCT(ZR) = 0 = ZR**3 - ZR**2 - A                       EM   650
C           USING THE NEWTON-RAPHSON METHOD                            EM   651
60    A = .666666*(3.*F/(BETPSQ*UU*S*ZB**3) -.2222222)                EM   652
      ZR = .75                                                         EM   653
      DO 70 I = 1,NITRAT                                               EM   654
      FCT = ZR**3 - ZR**2 - A                                          EM   655
      DFCT = 3.*ZR**2 -2.*ZR                                           EM   656
      ZRNU = ZR - FCT/DFCT                                             EM   657
      ERR = ABS(ZRNU-ZR)                                               EM   658
      IF(ERR.LE.ERLMT)  GO TO 80                                       EM   659
      ZR = ZRNU                                                        EM   660
70    CONTINUE                                                         EM   661
      DH = ZR*ZB                                                       EM   662
      WRITE(6,1) I,ERLMT,HTINV,HTPS,DH                                 EM   663
      GO TO 90                                                         EM   664
80    DH = ZR*ZB                                                       EM   665
      DH = AMIN1(DHMAX,DH)                                             EM   666
      IF(DEBUG.EQ.YES) WRITE(6,2) I,HTINV,HTPS,ZB,DH                   EM   667
90    PENTFR = 1.5 - 1./ZR                                             EM   668
      PENTFR = AMAX1(0.0,PENTFR)                                       EM   669
      PENTFR = AMIN1(1.0,PENTFR)                                       EM   670
100   RETURN                                                           EM   671
C                                                                      EM   672
1     FORMAT(1H0,10X,61HCONVERGENCE TO ROOTS OF PLUME RISE EQUATION NOT EM   673
     X ACHIEVED IN   ,I4,12H ITERATIONS  ,/,10X,5F12.3)                EM   674
2     FORMAT(1H ,10X,8HITERAT = ,I4,9H  HTINV = ,F6.1,9H  HTPS  =, F6.1,EM   675
     1  7H  ZB =   ,F6.1,   6H DH =   ,F6.1 )                          EM   676
      END                                                              EM   677


      FUNCTION DISTAN(IPT,TS,TE)                                       EM   678
C                                                                      EM   679
C        DETERMINES THE DISTANCE ALONG THE TRAJECTORY                  EM   680
C        BETWEEN TIMES *TS* AND *TE*.                                  EM   681
C                                                                      EM   682
      COMMON /WIND/  T(100),  V(100),  TM(100),  NPTS                  EM   683
      COMMON /LINKS/ SEGLEN(100),  RADISQ(100),  RADMUL,  PDISMX(6)    EM   684
      IF(TE.LT.T(NPTS))  GO TO 5                                       EM   685
      DISTAN = 1000000.                                                EM   686
      RETURN                                                           EM   687
5     IP1 = IPT + 1                                                    EM   688
      IF(T(IP1).LT.TE)  GO TO 10                                       EM   689
      DISTAN = (TE - TS)*V(IPT)                                        EM   690
      RETURN                                                           EM   691
10    DX = (T(IP1) - TS)*V(IPT)                                        EM   692
15    I1 = IP1                                                         EM   693
      IP1 = IP1 + 1                                                    EM   694
      IF(T(IP1) - TE)  20,25,25                                        EM   695
20    DX = DX + SEGLEN(I1)                                             EM   696
      GO TO 15                                                         EM   697
25    DX = DX + (TE - T(I1))*V(I1)                                     EM   698
      DISTAN = DX                                                      EM   699
```

CORRECTION

After EM 670, insert

```
    IF (PENTFR.LT. .1) PENTFR=0
```

```
      RETURN
      END


      FUNCTION ERF(X)
C
C        RATIONAL APPROXIMATIIION FOR THE ERROR FUNCTION FROM THE NBS
C        HANDBOOK OF MATHEMATICAL FUNCTIONS (AMS-55)  PG. 299.
C        MAX ERROR  =  2.5 * 10**-5
C
      DATA A1, A2, A3, P / .3480242, -.0958798, .7478556, .47047 /
      T = ABS(X)
      IF(T-5.)  2, 2, 10
2     V = 1./(1.+P*T)
      Q = 1. - ((((A3*V)+A2)*V+A1)*V)*EXP(-T*T)
      GO TO 20
10    Q = 1.000000
20    ERF = SIGN(Q,X)
      RETURN
      END


      SUBROUTINE GESTAB(TIME,ISTAB,DTNEXT,NXSTAB)
C
C        DETERMINES STABILITY CLASS
C
C         ISTAB  = STABILITY CLASS AT TIME SPECIFIED

C         DTNEXT = TIME TO NEXT STABILITY CLASS CHANGE
C         NXSTAB = STABILITY CLASS AT NEXT UPDATE TIME
C
      COMMON /PASDAL/  TMSTAB(25),  KSTABL(25),  ISTBMX
      IF(TIME.LT.TMSTAB(ISTBMX))  GO TO 10
5     ISTAB = KSTABL(ISTBMX)
      DTNEXT = 1000000.
      RETURN
10    CONTINUE
      IMAX = ISTBMX - 1
      DO 15 I = 1,IMAX
      IP = I + 1
      IF( (TIME.GE.TMSTAB(I)) .AND. (TIME.LT.TMSTAB(IP)))  GO TO 20
15    CONTINUE
      GO TO 5
20    ISTAB = KSTABL(I)
      NXSTAB = KSTABL(IP)
      DTNEXT = TMSTAB(IP) - TIME
      RETURN
      END
      SUBROUTINE GRIDIT(LUGRID,KOK)
C
C        *GRIDIT* READS THE RAPS/ST.LOUIS EMISSIONS GRID
C                 DISCRIPTION FILE FROM LUGRID.
C                 IT STORES ONLY THE GRID SQUARE IDENTIFIERS AND
C                 CORRESPONDING AREAS FOR USE IN -AREAEM*.
C
C     RAPS FILE DESCRIPTION
C           NGRID
C           ID  IAREA  UTMH  UTMV  STATE  COUNTY  HITE  RO
C           ID ... REPEATED NGRID TIMES
C
C
      DIMENSION   IWORK(16000),   IAREA(2000)
      COMMON /ANSWER/  YES,RNEG,     IYES,NO,          SMALL
      COMMON /DEGRID/  ID(2000),     AREA(2000),      NGRID
      COMMON /WORKER/  IDATE,    NG,   IHR,  WORK(22440)
      EQUIVALENCE(IWORK,WORK)
      EQUIVALENCE(IAREA,WORK(16001))
      DATA NUMREC, NWPREC, NVAR /32,510, 8/
      DATA ALL, LOUT /4HALL , 6/
      DEBUG = RNEG
      IREC = 1
      K1 = 0
      K2 = NWPREC - 1
C
C       ON THE UNIVAC REPLACE BUFFER IN AND IF UNIT WITH
C     READ(LUGRID,ERR=220,END=210) IHR,(WORK(JJ),JJ=1,K2)
C
      BUFFER IN (LUGRID,1) (IHR,WORK(K2))
120   IF(UNIT(LUGRID))     130,   210,   220
130   NGRID = IHR
      IF(DEBUG.NE.ALL)  GO TO 140
      WRITE(LOUT,5) NGRID
      WRITE(LOUT,6) (IWORK(I),I=K1,K2)
140   CONTINUE
      K2MAX = NVAR*NGRID
      DO 170 IREC = 2,NUMREC
      K1 = K1 + NWPREC
      K2 = K2 + NWPREC
      IF(IREC.EQ.NUMREC)  K2 = MIN0(K2,K2MAX)
C        ON THE UNIVAC REPLACE BUFFER IN AND IF(UNIT) WITH
C     READ(LUGRID,ERR=220,END=210) (WORK(JJ),JJ=K1,K2)
C
      BUFFER IN (LUGRID,1)  (WORK(K1),WORK(K2))
150   IF(UNIT(LUGRID))   160,  210,  220
160   IF(DEBUG.NE.ALL)  GO TO 170
      WRITE(LOUT,6)  IREC
      WRITE(LOUT,6) (IWORK(I),I=K1,K2)
170   CONTINUE
      K1 = 1
      K2 = 2
      DO 180 J = 1,NGRID
      ID(J) = IWORK(K1)
      IAREA(J) = IWORK(K2)
      AREA(J) = FLOAT(IAREA(J))
      K1 = K1 + NVAR
      K2 = K2 + NVAR
180   CONTINUE
      IF( (DEBUG.EQ.YES).OR.(DEBUG.EQ.ALL) )  GO TO 190
      RETURN
190   WRITE(LOUT,5)  NGRID
      WRITE(LOUT,7)  (ID(J), IAREA(J), J=1,NGRID)
      RETURN
210   KOK = 9
      WRITE(LOUT,4) KOK
      RETURN
220   KOK = 99
      WRITE(LOUT,3) KOK
      RETURN
3     FORMAT (44H1PARITY ERROR ON GRID FILE  --  JOB ABORTED  ,I8)
4     FORMAT (46H1UNEXPECTED EOF IN GRID FILE  --  JOB ABORTED   ,I8)
5     FORMAT (1H0,10H    NGRID=  ,I6)
6     FORMAT (1H ,3(8I5,3X))
7     FORMAT (1H ,10(2I4,4X))
      END



      SUBROUTINE LOCATE(IPT,XP,YP,PDIST,TPASS,KFLAG,ISTAB)
C
C         LOCATE DETERMINES WHETHER OR NOT A POINT SOURCE
C         IS PASSED ON A TRAJECTORY SEGMENT AND WHETHER OR
C         NOT THE POINT SOURCE IS CLOSE ENOUGH TO BE CONSIDERED.
C
C           IPT   =   TRAJECTORY NODE INDEX
C           XP    =   X COORDINATE OF POINT SOURCE IN KM
C           YP    =   Y COORDINATE OF POINT SOURCE IN KM
C           PDIST =   PERPENDICULAR DISTANCE TO POINT SOURCE IN KM
C           TPASS =   TIME SOURCE IS PASSED
C           KFLAG =   FLAG WHICH IS POSITIVE IF SOURCE MEETS ALL
C                     CRITERIA FOR CONSIDERATION ON TRAJECTORY SEGMENT I
C           ISTAB =   STABILITY CLASS
C
      DIMENSION DX(2), DY(2), DIST(2), ALFA(2), BETA(2)
C
      COMMON  /WIND/  T(100),  V(100),  TH(100),  NPTS
      COMMON /LINKS/  SEGLEN(100),  RADISQ(100),  RADMUL, PDISMX(6)
      COMMON  /TRAJ/  P(2,100)
      COMMON /ANGLES/ GAMA1(100),  GAMA2(100),  ETA1(100),  ETA2(100)
      DATA PIE20 /6.283185307/
C
      DX(1) = XP - P(1,IPT)
      DY(1) = YP - P(2,IPT)
      DISTSQ = DX(1)**2  +  DY(1)**2
C         CHECK DISTANCE TO POINT SOURCE RELATIVE TO RADMUL*SEGMENT LENG
      IF(DISTSQ.LE.RADISQ(IPT))  GO TO 10
      KFLAG = -1
      RETURN
10    BETA(1) = ATAN2(DY(1),DX(1))
```

```
      IF(BETA(1).LT. 0.0)  BETA(1) = BETA(1) + PIE20
      ALFA(1) = BETA(1) - TH(IPT)
      IF(ALFA(1).LT. 0.0)  ALFA(1) = ALFA(1) + PIE20
      IF(ALFA(1).GT. PIE20)  ALFA(1) = ALFA(1) - PIE20
      DIST(1) = SQRT(DISTSQ)
      PDIST = DIST(1)*SIN(ALFA(1))
C         CHECK PERPENDICULAR DISTANCE
      IF(ABS(PDIST).LT.PDISMX(ISTAB))  GO TO 20
      KFLAG = - 1
      RETURN
20    CONTINUE
      IP1 = IPT + 1
C         CHECK BISECTOR ANGLES
      DX(2) = XP - P(1,IP1)
      DY(2) = YP - P(2,IP1)
      BETA(2) = ATAN2(DY(2),DX(2))
      IF(BETA(2).LT. 0.00)   BETA(2) = BETA(2) + PIE20
      ALFA(2) = BETA(2) - TH(IPT)
      IF(ALFA(2) .LT. 0.00)  ALFA(2) = ALFA(2) + PIE20
      IF(ALFA(2) .GT. PIE20)  ALFA(2) = ALFA(2) - PIE20
      KFLAG = - 1
C
      IF(ETA1(IPT) .GE. GAMA1(IPT))  GO TO  60
C
40    IF(PDIST .LT. 0.00)   GO TO 50
      IF((ALFA(1).LE.GAMA1(IPT)) .AND. (ALFA(2).GT.ETA1(IPT))) KFLAG = 1
      GO TO 100
50    IF((ALFA(1).GE.GAMA2(IPT)) .AND. (ALFA(2).LT.ETA2(IPT))) KFLAG = 2
      GO TO 100
60    IF(PDIST .GE. 0.00)   GO TO 70
      IF((ALFA(1).GE.GAMA2(IPT)) .AND. (ALFA(2).LT.ETA2(IPT))) KFLAG = 3
      GO TO 100
70    IF((ALFA(1).LE.GAMA1(IPT)) .AND. (ALFA(2).GT.ETA1(IPT))) KFLAG = 4
100   IF(KFLAG.LT.0)  RETURN
      DAXIAL = DIST(1)*COS(ALFA(1))
      TPASS = T(IPT) + DAXIAL/V(IPT)
      TPASS = AMAX1(T(IPT),TPASS)
      TPASS = AMIN1(T(IP1),TPASS)
      RETURN
      END


      SUBROUTINE PARTIT(X,HT,HTINVR,HTINTF,NOSTAT,ISTAB,TALL,PENTFR
     1                                                        ,VFRACT)
C
C        *PARTIT*  PERFORMS THE VERTICAL INTEGRATION OF THE GAUSSIAN
C        DISTRIBUTION OVER VERTICAL CELLS DEFINED BY HTINTF.  IT ASSUMES
C        A REFLECTION AT THE SURFACE AND AT THE INVERSION BASE.
C        THE MAXIMUM VALUE OF THE INTEGRAL HAS BEEN NORMALIZED TO
C        ONE MINUS THE INVERSION PENETRATION FRACTION.
C
C     INPUTS
C        X =       DOWNWIND DISTANCE (KM)
C        HT =      SOURCE EFFECTIVE HEIGTH (METERS)
```

```
C        HTINVR = HEIGHT OF INVERSION (METERS)
C        HTINTF = HEIGHTS OF CELL INTERFACES  (METERS)
C        NOSTAT = NUMBER OF CELLS
C        ISTAB =  STABILTY (1-6 CORRESPONDING TO A-F)
C        TALL  =  TALL STACK INDICATOR (EQUALS YES OR NO)
C        PENTFR = INVERSION PENETRATION FRACTION (0 TO 1)
C     OUTPUT
C        VFRACT = VECTOR OF FRACTIONS OF NORMAL DISTRIBUTION
C                 IN EACH CELL (DEFINED BY HTINTF)
C
      DIMENSION  ZE(6),   HTINTF(1),    VFRACT(1)
      COMMON /SIGMAS/  NDY,   XDY(4),   SIGYD(4,6),  NDZ, XDZ(12),
     1                 SIGZD(12,6),     NCAT
      COMMON /SIGTAL/ ATALL(6), BTALL(6), CTALL(6), DTALL(6)
      DATA SQRT2 /1.414213562/
      DATA YES, RNEG, DZINV /4HYES ,4HNO  ,10./
C
      DO 10 I = 1,NOSTAT
      VFRACT(I) = 0.000
10    CONTINUE
      CPENTR = 1.000 - PENTFR
      NSP1 = NOSTAT + 1
      S = -1.00
      IF(HT.GE.HTINTF(NSP1))  GO TO 200
C
      IF(TALL.EQ.YES)  GO TO 5
      SIGZ = TRPLOG(X,XDZ,SIGZD(1,ISTAB),NDZ,1)
      GO TO 8
5     SIGZ = CTALL(ISTAB)*X**DTALL(ISTAB)
8     SIGZ2 = SIGZ*SQRT2
      DO 15 I = 2,NSP1
      IF(HTINVR.LT.HTINTF(I))  GO TO 20
15    CONTINUE
20    IMAX = I - 1
C     IF(HTINVR - HT) 120,120,25
      IF(PENTFR.GT. 0.99) GO TO 120
C     -------- SOURCE HEIGHT BELOW INVERSION HEIGHT
25    DO  70 NP = 1,2
      NBELOW = 0
      DO 30 J = 1,NSP1
      ZE(J) = HTINTF(J) + HT*S
      IF(ZE(J).LT.0.00)   NBELOW = NBELOW + 1
30    CONTINUE
      HTLMT = HTINVR + HT*S
      I = 1
      IF(NBELOW.LT.1)  GO TO 50
      DO 40 I = 1,NBELOW
      ZF = ABS(ZE(I)/SIGZ2)
      ZN = ZE(I+1)/SIGZ2
      ZN = AMIN1(ZN,0.0000)
      ZN = ABS(ZN)
      VFRACT(I) = VFRACT(I) + .5*(ERF(ZF) - ERF(ZN))
40    I=NBELOW
50    CONTINUE
      DO 60 J = I,NOSTAT
      ZN = ZE(J)/SIGZ2                                                  EM   958
      ZN = AMAX1(ZN,0.000)                                              EM   959
      ZF = ZE(J+1)                                                      EM   960
      IF (ZF.LT.HTLMT)  GO TO 55                                        EM   961
      ZF = 10000.                                                       EM   962
      VFRACT(K) = VFRACT(K) + .5*(ERF(ZF) - ERF(ZN))                    EM   963
      GO TO 65                                                          EM   964
55    ZF = ZF/SIGZ2                                                     EM   965
      VFRACT(J) = VFRACT(J) + .5*(ERF(ZF) - ERF(ZN))                    EM   966
60    CONTINUE                                                          EM   967
65    S = 1.00                                                          EM   968
70    CONTINUE                                                          EM   969
      DO 80 I = 1,NOSTAT                                                EM   970
      S = S - VFRACT(I)                                                 EM   971
80    CONTINUE                                                          EM   972
C     --------- DISTRIBUTE REMAINDER EVENLY BELOW THE INVERSION         EM   973
      SPMETR = S/HTINVR                                                 EM   974
      DO 90 I = 1,IMAX                                                  EM   975
      HT1 = AMIN1(HTINVR,HTINTF(I+1))                                   EM   976
      ZF = SPMETR*(HT1 - HTINTF(I))                                     EM   977
      VFRACT(I) = (VFRACT(I) + ZF)*CPENTR                               EM   978
90    CONTINUE                                                          EM   979
      HT1 = HTINTF(IMAX+1) - HTINVR                                     EM   980
      IF(HT1 - DZINV) 100,100,110                                       EM   981
100   IMAX = MINO(IMAX+1,NOSTAT)                                        EM   982
      WRITE(6,7)                                                        EM   983
7     FORMAT(1H ,* INVERSION HEIGHT CLOSE TO TOP OF CELL --  PENETRATIONEM   984
     1  FRACTION PLACED IN NEXT CELL *)                                 EM   985
110   VFRACT(IMAX) = VFRACT(IMAX) + PENTFR                              EM   986
      RETURN                                                            EM   987
C                                                                       EM   988
120   CONTINUE                                                          EM   989
C       -------- SOURCE HEIGHT ABOVE INVERSION HEIGHT                   EM   990
      DO 130 J = 2,NSP1                                                 EM   991
      IF(HT.LT.HTINTF(J))  GO TO 140                                    EM   992
130   CONTINUE                                                          EM   993
140   JMAX = J - 1                                                      EM   994
      VFRACT(JMAX) = VFRACT(JMAX) +  1.00                               EM   995
      RETURN                                                            EM   996
C       -------- SOURCE HEIGHT ABOVE TOP OF MESH                        EM   997
200   WRITE(6,1)  HT,HTINTF(NSP1)                                       EM   998
      VFRACT(NOSTAT) = 1.00                                             EM   999
      RETURN                                                            EM  1000
1     FORMAT(1H0, 25HEFFECTIVE SOURCE HEIGHT = ,F10.1,/,                EM  1001
     118H MESH HEIGHT     =  F10.1,  //, 20H SOURCE IN TOP CELL       ) EM  1002
      END                                                               EM  1003
```

CORRECTION

Insert after EM979

IF(VFRACT(I).LT. 1.E-4) VFRACT(I)=0.

```
      SUBROUTINE PLUMAS(IPT,TPASS,PDIST,PLUMFR,ISTAB,DXFREZ,TALL,KPLAG)
C
C       *PLUMAS* PERFORMS THE LATERAL INTERGRATION OF THE GUASSIAN
C       DISTRIBUTION TO DETERMINE THE PORTION OF THE PLUME'S MASS
C       REMAINING IN THE LAGRANGIAN AIR PARCEL AFTER DTFREZ MINUTES.
C
C       IPT      = TRAJECTORY NODE INDEX
C       TPASS    = TIME AT WHICH POINT SOURCE WAS PASSED
C       PDIST    = PERPENDICULAR DISTANCE TO SOURCE
C       PLUMFR   = RELATIVE PORTION OF PLUME'S MASS REMAINING IN PARCEL
C                  (  0.< PLUMFR < 1.0  )
C       ISTAB    = STABILITY (1-6 CORRESPONDING TO A-F) RETURNED
C       DXFREZ   = DOWNWIND DISTANCE TRAVELLED IN DTFREZ CORRECTED
C                  FOR STABILITY CLASS CHANGES.
C       TALL     = TALL STACK (GT 100 METERS) FLAG = YES OR NO
C       KPLAG    =  1       LATERAL DIFFUSION OUT OF PARCEL IS DOMINANT
C       KPLAG    =  2       LATERAL DIFFUSION INTO PARCEL IS DOMINANT
C       KPLAG    = -1       PLUME'S LATERAL SPREAD IS TOO SLOW TO REACH
C                           PARCEL   --  SOURCE SHOULD BE IGNORED ENTIREL
C
      COMMON /SIGMAS/  NDY,  XDY(4),  SIGYD(4,6),  NDZ,  XDZ(12),
     1                 SIGZD(12,6),  NCAT
      COMMON /ANSWER/  YES, RNEG, IYES, NO, SMALL
      COMMON /SIGTAL/  ATALL(6), BTALL(6), CTALL(6), DTALL(6)
      COMMON /PARCEL/ PWIDTH, HPWIDT, PLENTH, PAREA, DTFREZ, SIGEDG,
     1                ZEE(5), HTINTF(6),  NOSTAT,  ROAIR
      DATA SQRT2,   SMALL   /1.414213562,  .020/
C
      KPLAG = - 1
      IP1 = IPT + 1
      PDIST = ABS(PDIST)
      TFREEZ = TPASS + DTFREZ
      DXFREZ = DISTAN(IPT,TPASS,TFREEZ)
      VBAR = DXFREZ/DTFREZ
      CALL GESTAB(TPASS,ISTAB,DTSTCH,NXSTAB)
      IF(DTSTCH.GT.DTFREZ-5) GO TO 100
C
C       ADJUST DXFREZ TO ACCOUNT FOR STABILITY CLASS CHANGE
C
      TSTCH = TPASS + DTSTCH
      DXSTCH = DISTAN(IPT,TPASS,TSTCH)
      IF(TALL.EQ.YES)   GO TO 80
      SIGSCH = TRPLOG(DXSTCH,XDY,SIGYD(1,ISTAB),NDY,1)
      DXVIRT = TRPLOG(SIGSCH,SIGYD(1,NXSTAB),XDY,NDY,1)
      GO TO 90
80    SIGSCH = ATALL(ISTAB)*(DXSTCH**BTALL(ISTAB))
      DXVIRT = (SIGSCH/ATALL(NXSTAB))**(1./BTALL(NXSTAB))
90    DXDIFF = DXSTCH - DXVIRT
      DXFREZ = DXFREZ - DXDIFF
      ISTAB = NXSTAB
C
C       INTEGRATE PLUMES FROM SOURCES WITHIN PARCEL WIDTH
C
100   CONTINUE
      IF(TALL.EQ.YES)  GO TO 110
      SIGY = TRPLOG(DXFREZ,XDY,SIGYD(1,ISTAB),NDY,1)
      GO TO 115
110   SIGY = ATALL(ISTAB)*(DXFREZ**BTALL(ISTAB))
115   SIGYSP = SIGY*SQRT2
C
      IF(PDIST.GT.HPWIDT)  GO TO 120
C
      D1 = (HPWIDT - PDIST)/SIGYSP
      D2 = (HPWIDT + PDIST)/SIGYSP
      PLUMFR = 0.50*(ERF(D1) + ERF(D2))
      KPLAG = 1
      RETURN
C
C       INTEGRATE PLUMES FROM SOURCES OUTSIDE PARCEL WIDTH
C       AND ADJUST TPASS TO ACCOUNT FOR THE TIME DELAY BEFORE
C       INTERSECTION OF THE PLUME.
C
120   CONTINUE
      IF(PDIST.LT. HPWIDT + SMALL)  GO TO 140
      SIGY2 = (PDIST - HPWIDT)/SIGEDG
      IF(TALL.EQ.YES)  GO TO 125
      DXEDG = TRPLOG(SIGY2,SIGYD(1,ISTAB),XDY,NDY,1)
      GO TO 130
125   DXEDG = (SIGY2/ATALL(ISTAB))**(1./BTALL(ISTAB))
130   IF(DXEDG .GT.DXFREZ)   RETURN
      TCROSS = TFREEZ -(DXFREZ - DXEDG)/VBAR
      TPASS = AMAX1(TPASS,TCROSS)
140   CONTINUE
      D1 = (PDIST - HPWIDT)/SIGYSP
      D2 = (PDIST + HPWIDT)/SIGYSP
      PLUMFR = 0.50*(ERF(D2) - ERF(D1))
      KPLAG = 2
      RETURN
      END



      SUBROUTINE PONTEM(LUPONT,IPUNCH,KOK,NOSUMS)
C
C          *PONTEM* DETERMINES THE POINT SOURCE EMISSIONS
C          HISTORY ALONG A SPECIFIED TRAJECTORY.
C
C          THIS IS THE RAPS VERSION OF PONTEM
C
      DIMENSION          VFRACT(5),      IWORK(200),     WORK1(200),
     1                   WORK2(5,200),   FULRAT(7,200), WORK4(7,200)
      COMMON /INPUTS/    TITLE(20),      IDATE(10),      NCASE
      COMMON /WORKER/    IXT(3),         WORK(22440)
      COMMON /ANSWER/    YES,RNEG,       IYES,NO,        SMALL
      COMMON /LABLIN/    NAMIN(11),      NFLXIN,         WTMOLI(11),
     1                   ADJUST(11)
      COMMON /LABOUT/    NAMOUT(7),      NFXOUT,         WTMOLO(7)
      COMMON /TABLES/    THOUR(24)
      COMMON /PASSES/    TPASS(200),     VFR(5,200),     PSMASS(7,200),
     1                   PMFR(200),      PSPD(200),      NKPS
```

```
      COMMON /TRAJ/    P(2,100)
      COMMON /WIND/    T(100),         V(100),        TH(100),
     1                 NPTS
      COMMON /LINKS/   SEGLEN(100),    RADISQ(100),   RADMUL,
     1                 PDISMX(6)
      COMMON /PARCEL/  PWIDTH,         HPWIDT,        PLENTH,
     1                 PAREA,          DTFHEZ,        SIGEDG,
     2                 ZEE(5),         HTINTF(6),     NOSTAT,
     3                 ROAIR
      COMMON /TEMPS/   TMTEMP(25),     TEMPSF(25),    NTEMP,ITEMP
      COMMON /MXHITE/  TMXHIT(25),     HTMIXL(25),    NMIXL,IMIXL
      COMMON /PSDAT/   PSXY(2,505),    PSRAT(7,505),  PSTEMP(505),
     1                 PSFLOW(505),    PSHITE(505),   NPS
      COMMON /ORIGIN/  UTMXOR,         UTMYOR
      COMMON /ANGLES/  GAMA1(100),     GAMA2(100),    ETA1(100),
     1                 ETA2(100)
      COMMON /SIGMAS/  NDY,            XDY(4),        SIGYD(4,6),
     1                 NDZ,            XDZ(12),       SIGZD(12,6),NCAT
      EQUIVALENCE(IWORK,WORK(1))
      EQUIVALENCE(WORK1,WORK(201))
      EQUIVALENCE(WORK2,WORK(401))
      EQUIVALENCE(WORK4,WORK(1401))
      EQUIVALENCE(FULRAT,WORK(2801))
      DATA VOXFRC, KPSMAX  /0.3333,  200/
      DATA LOUT, LPUNCH /6,1/
C
      DEBUG = RNEG
      KOK = 100
      NM1 = NPTS - 1
      KPS = 0
      IWAS = 0
      DTGZ = .0137
      DO 50 I = 1,NPTS
      P(1,I) = P(1,I) - UTMXOR
      P(2,I) = P(2,I) - UTMYOR
   50 CONTINUE
      CALL SEGMET(P)
C
      CALL STACKS(T(1),LUPONT,KOK,IWAS,NDSUMS)
      IWAS = 1
C
      IF(KOK .LT. 0)   RETURN
C
      DO 200 IL = 1,NM1
      IN = IL + 1
C
      CALL GESTAB(T(IL),ISTAB,DX,N)
C
      HRFRAC = PLENTH/(V(IL)*60.)
      HTV =HTMIXL(IMIXL)
      TA = TEMPSF(ITEMP)
C
C       LOCATE POINT SOURCES AND INTEGRATE THEIR PLUMES
C
      DO 100 JS = 1,NPS
```

```
C
      CALL LOCATE(IL,PSXY(1,JS),PSXY(2,JS),PDIST,TPASIT,KFLAG,ISTAB)
C
      IF(KFLAG.LT.0)   GO TO 100
      IF(DEBUG.EQ.YES)
     1 WRITE(LOUT,21) JS,PSXY(1,JS),PSXY(2,JS),PDIST,TPASIT,ISTAB,KFLAG
C
      TALL = RNEG
      HTPS = PSHITE(JS)
      IF(HTPS.GE.100.)  TALL = YES
      CALL PLUMAS(IL,TPASIT,PDIST,PMFRAC,ISTAB,DXFREZ,TALL,KPLAG)
C
      IF(KPLAG.LT.0)   GO TO 100
      IF(DEBUG.EQ.YES)
     1 WRITE(LOUT,21)JS,TPASIT,PDIST,PMFRAC,DXFREZ,ISTAB,KPLAG
      DO 70 JR = 1,NFXOUT
      IF(PSRAT(JR,JS).GT.0.0)  GO TO 75
70    CONTINUE
      GO TO 100
75    KPS = KPS + 1
      IF(KPS.GT.KPSMAX)   GO TO 510
      TPASS(KPS) = TPASIT
      PMFR(KPS) = PMFRAC
      PSPD(KPS) = PDIST
      F = PSFLOW(JS)
      TS = PSTEMP(JS)
      DH = 0.0
      IF(TS.LE.TA)  GO TO 90
      IF(F.LT.SMALL)  GO TO 90
      U = V(IL)*16.6667
      CALL DHPLUM(TA,TS,U,ISTAB,F,HTV,HTPS,DH,PF,DTDZ)
      IF(DEBUG.EQ.YES) WRITE(LOUT,21) JS,HTEFF,HTPS,HTV
90    HTEFF = DH + HTPS
      DXFREZ = VDXFRC*DXFREZ
C
      CALL PARTIT(DXFREZ,HTEFF,HTV,HTINTF,NOSTAT,ISTAB,TALL,PF,VFRACT)
C
      IF(DEBUG.EQ.YES) WRITE(LOUT,20)  KPS, VFRACT
      CALL XMIT(NOSTAT,VFRACT,VFR(1,KPS))
      DO 95 K = 1,NFXOUT
      FULRAT(K,KPS) = PSRAT(K,JS)
      PSMASS(K,KPS) = PMFRAC*HRFRAC*PSRAT(K,JS)
95    CONTINUE
100   CONTINUE
C
      IF(IN.EQ.NPTS)  GO TO 210
      IF(ITEMP.GE.NTEMP)  GO TO 120
      IF(T(IN).LT.TMTEMP(ITEMP+1)-SMALL)  GO TO 120
      ITEMP = ITEMP + 1
120   CONTINUE
      IF(IMIXL.GE.NMIXL) GO TO 130
      IF(T(IN).LT.TMXHIT(IMIXL+1)-SMALL)  GO TO 130
      IMIXL = IMIXL + 1
130   CONTINUE
      CALL STACKS(T(IN),LUPONT,KOK,IWAS,NOSUMS)
      IF(KOK .LT. 0)  GO TO 210
 200  CONTINUE
C
 210  CONTINUE
      IF(KPS .LE. 0)  GO TO 520
C
C        ORDER PASSING ARRAYS CHRONOLOGICALLY
C
      DO 220 N = 1,KPS
      IWORK(N) = N
 220  CONTINUE
      DO 230 N = 1,KPS
      M = KPS + 1 - N
      CALL FMINF(TPASS(N),M,FMIN,NMIN)
      NMIN = NMIN + N - 1
      J = IWORK(N)
      IWORK(N) = IWORK(NMIN)
      IWORK(NMIN) = J
      XX = TPASS(N)
      TPASS(N) = FMIN
      TPASS(NMIN) = XX
 230  CONTINUE
C
      CALL XMIT(KPS, PMFR, WORK1)
      N = 5*KPS
      CALL XMIT(N,VFR,WORK2)
      DO 240 N = 1,KPS
      J = IWORK(N)
      PMFR(N) = WORK1(J)
      DO 240 K = 1,NOSTAT
      VFR(K,N) = WORK2(K,J)
 240  CONTINUE
C
      CALL XMIT(-NFXOUT,0,0,WORK1)
      M = 7*KPS
      CALL XMIT(M,PSMASS,WORK4)
      DO 250 N = 1,KPS
      J = IWORK(N)
      DO 250 K = 1,NFXOUT
      PSMASS(K,N) = WORK4(K,J)
      WORK1(K) = WORK1(K) + PSMASS(K,N)
 250  CONTINUE
      CALL XMIT(KPS,PSPD,SEGLEN)
      CALL XMIT(M,FULRAT,WORK4)
      DO 260 N = 1,KPS
      J = IWORK(N)
      PSPD(N) = SEGLEN(J)
      DO 260 K = 1,NFXOUT
      FULRAT(K,N) = WORK4(K,J)
 260  CONTINUE
C
C     OUTPUTS
C    1. ACTUAL SOURCE RATES, FRACTION INCLUDED, AND PERPENDICULAR DISTAN
C    2. ABSOLUTE MASS (MOLES) AND VERTICAL DISTRIBUTION
C    3. MOLE FRACTION-METER OF PARCEL AIR
```

```
C
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,5)  (NAMOUT(K),K=1,NFXOUT)
      DO 310 J = 1,KPS
      IF(MOD(J,25).NE.0)  GO TO 300
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,5)  (NAMOUT(K),K=1,NFXOUT)
300   WRITE(LOUT,6) TPASS(J),PMFR(J),PSPD(J),(FULRAT(K,J),K=1,NFXOUT)
310   CONTINUE
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,7)  (NAMOUT(K),K=1,NFXOUT)
      WRITE(LOUT,8)  (HTINTF(K),HTINTF(K+1),K=1,NOSTAT)
      DO 330 J = 1,KPS
      IF(MOD(J,18).NE.0)  GO TO 320
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,7)  (NAMOUT(K),K=1,NFXOUT)
      WRITE(LOUT,8)  (HTINTF(K),HTINTF(K+1),K=1,NOSTAT)
320   WRITE(LOUT,9)  TPASS(J), (PSMASS(K,J),K=1,NFXOUT)
      WRITE(LOUT,10) (VFR(K,J),K=1,NOSTAT)
330   CONTINUE
      WRITE(LOUT,11) (WORK1(K),K=1,NFXOUT)
C     CONVERT TO MOLE FRACTION*METER OF AIR PARCEL
      A = 28.97/(ROAIR*PAREA*1.E6)
      DO 340 J = 1,KPS
      DO 340 K = 1,NFXOUT
      PSMASS(K,J) = A*PSMASS(K,J)
340   CONTINUE
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,12)  (NAMOUT(K),K=1,NFXOUT)
      DO 360 J = 1,KPS
      IF(MOD(J,25).NE.0)  GO TO 350
      CALL NEWPAG(TITLE,0,JDATE)
      WRITE(LOUT,12)  (NAMOUT(K),K=1,NFXOUT)
350   WRITE(LOUT,9)  TPASS(J),  (PSMASS(K,J),K=1,NFXOUT)
      IF(IPUNCH.NE.IYES)  GO TO 360
      WRITE(LPUNCH,14) J, TPASS(J), (PSMASS(K,J),K=1,NFXOUT)
      WRITE(LPUNCH,15) J, (VFR(K,J),K=1,NOSTAT)
360   CONTINUE
370   RETURN
C
510   WRITE(LOUT,22)
      KOK = -100
      RETURN
520   WRITE(LOUT,23)
      RETURN
C
5     FORMAT(47H0POINT SOURCE EMISSION RATES   --   MOLES/HOUR   ,
     1 //,5X,5HTIME ,5X,20HFRACTION    DISTANCE    ,7(5X,A4,3X))
6     FORMAT(1H0,F10.2,F10.4,F11.3,4X,7E12.4)
7     FORMAT(66H0POINT SOURCE EMISSIONS   --   MOLES AND VERTICAL DISTRI
     1BUTION        ,//,8X,7HSPECIES,6X,7(5X,A4,3X))
8     FORMAT(1H ,7X,10HELEVATIONS,4X,5(F5.0,1H-,F4.0,2X),/,5X,4HTIME)
9     FORMAT(1H0,F10.2,10X,7E12.4)
10    FORMAT(1H ,16X,7F12.4)
11    FORMAT(1H ,20X,7(2X,10H========== )//,5X,12HTOTAL MOLES 4X,7E12.4)
```

```
12    FORMAT(49H0POINT SOURCE EMISSIONS  --  MOLE FRACTION-METERS,
     1  //,5X,5HTIME ,11X,7(5X,A4,3X))
14    FORMAT(13HPOINT SOURCES,I3,2X,F12.2,4E12.4,/,30X,3E12.4)
15    FORMAT(13X,I3,2X,5E12.4)
20    FORMAT(1H ,I5,6F12.4)
21    FORMAT(1H ,I5,4F12.4,2I5)
22    FORMAT(40H0TPASS ARRAY OVERFLOW  --  JOB ABORTED    )
23    FORMAT(46H0THERE ARE NO CLOSE ELEVATED POINT SOURCES      )
      END



      SUBROUTINE SEGMET(P)
C
C       SEGMET COMPUTES THE LENGTH AND BISECTOR ANGLES FOR EACH
C       SEGMENT OF THE TRAJECTORY.
C       SEGMET COMPUTES THE MAXIMUM RADII AND PERPENDICULAR DISTANCE
C       USED IN SELECTING POINT SOURCES.
C
      DIMENSION   P(2,100),  HTH(100)
      COMMON /WIND/   T(100),   V(100),   TH(100),   NPTS
      COMMON /ANSWER/ YES,RNEG, IYES,NO,  SMALL
      COMMON /LINKS/  SEGLEN(100),  RADISQ(100), RADMUL, PDISMX(6)
      COMMON /ANGLES/ GAMA1(100), GAMA2(100), ETA1(100), ETA2(100)
      COMMON /PARCEL/ PWIDTH, HPWIDT, PLENTH, PAREA, DTFREZ, SIGEDG,
     1                ZEE(5), HTINTF(6), NOSTAT,  ROAIR
      COMMON /SIGMAS/ NDY,XDY(4),SIGYD(4,6),NDZ,XDZ(12),SIGZD(12,6),NCAT
      DATA PIE05, PIE10, PIE15, PIE20 /1.570796327, 3.141592654,
     1     4.71238898,  6.283185307 /
      DATA  RADMUL  /2.5/
C
      DEBUG = RNEG
      VMAX = 0.0000
      HTH(1) = 0.0000
      NM1 = NPTS - 1
      DO 50 I = 1,NM1
      J = I + 1
      DSQ = (P(1,J) - P(1,I))**2   +   (P(2,J) - P(2,I))**2
      RADISQ(I) = DSQ*RADMUL
      SEGLEN(I) = SQRT(DSQ)
      THET = TH(J) - TH(I)
      HTH(J) = 0.50*THET
      IF(HTH(J).GT.PIE10)   HTH(J) = HTH(J) - PIE20
      IF(HTH(J).LT.-PIE10)  HTH(J) = HTH(J) + PIE20
C    (IE..... -PIE <  HALF THETA  <  PIE  )
      IF  (V(I).LE.VMAX)  GO TO 50
      VMAX = V(I)
50    CONTINUE
C
      DO 100 I = 1,NM1
      J = I + 1
      GAMA1(I) = PIE05 - HTH(I)
      GAMA2(I) = PIE15 - HTH(I)
      ETA1(I)  = PIE05 + HTH(J)
      ETA2(I)  = PIE15 + HTH(J)
      IF(GAMA1(I).GE.0.0)  GO TO 90
      GAMA1(I) = GAMA1(I) + PIE10
      GAMA2(I) = GAMA2(I) + PIE10
90    IF(ETA1(I).LT.PIE20) GO TO 100
      ETA1(I) = ETA1(I) - PIE10
      ETA2(I) = ETA2(I) - PIE10
100   CONTINUE
C
      DXMAX = VMAX*DTFREZ
      DO 150 I = 1,6
      SIGY = TRPLOG(DXMAX,XDY,SIGYD(1,I),NDY,1)
      PDISMX(I) = SIGY*SIGEDG
      PDISMX(I) = AMAX1(HPWIDT,PDISMX(I))
150   CONTINUE
C
      IF(DEBUG.NE.YES) RETURN
C
      WRITE(6,1)
      DEG = 180./PIE10
      DO 200 I = 1,NM1
      A = TH(I)*DEG
      B = HTH(I)*DEG
      C = HTH(I+1)*DEG
      D = GAMA1(I)*DEG
      E = GAMA2(I)*DEG
      F = ETA1(I)*DEG
      G = ETA2(I)*DEG
      WRITE(6,2) I,A,B,C,D,E,F,G
200   CONTINUE
      WRITE(6,3) (I,PDISMX(I),I=1,6)
      RETURN
1     FORMAT(1H0,12H ANGLES A-G      )
2     FORMAT(1H ,I5,7F10.2)
3     FORMAT(41H0MAXIMUM PERPENDICULAR DISTANCE CRITERIA     ,/
     1  ,37H0STABILITY CLASS    DISTANCE . (KM)                ,
     2  6(//10X,I1,10X,F6.3))
      END



      SUBROUTINE STACKS(TIME,LUP,KOK,IWAS,NDSUMS)
C
C         *STACKS* READS THE RAPS POINT SOURCE EMISSIONS FILE
C          AND FILLS THE ARRAYS IN COMMON BLOCK /PSDAT/ EACH HOUR.
C
C          PSXY    = UTM X & Y COORDINATES (KILOMETERS)
C          PSHITE  = STACK HEIGHTS     (METERS)
C          PSTEMP  = STACK EFFLUENT TEMPERATURE   (CENTIGRADE)
C          PSFLOW  = STACK EFFLUENT VOLUMETRIC FLOW RATE  (M**3/MIN)
C          PSRATE  = EMISSION RATES FOR VARIOUS SPECIES  (MOLES/HR)
C          NPS     = NUMBER OF POINT SOURCES  (FOR THE CURRENT HOUR)
C          TIME    = TIME IN MINUTES FROM MIDNIGHT
C          LUP     = LOGICAL UNIT OF POINT SOURCE EMISSIONS FILE
C          NDSUMS  = REQUEST REGIONAL TOTALS (YES OR NO)
C
```

```
C          RAPS FILE DESCRIPTION
C
C               IDAT NSTACKS UTMX UTMY HITE DIAM TEMP FLOW PART
C                            SOX  NOX  THC  CO   NRHC PARF OLEF
C                            AROM ALDE STACKID
C                            UTMX UTMY HITE .............
C                            ..........................
C                            .......... STACKID
C                            (REPEATED NSTACK TIMES PER HOUR)
C
      DIMENSION ISOURC(17,505)
      COMMON /ANSWER/  YES,RNEG,      IYES,NO,    SMALL
      COMMON /INPUTS/  TITLE(20),     JDATE(10),  NCASE
      COMMON /ORIGIN/  UTMXOR,        UTMYOR
      COMMON /LABLIN/  NAMIN(11),     NFLXIN, WT(11), ADJUST(11)
      COMMON /  WORKER/  IDAT,NSTACK,  SUM(510),  WORK(8585),
     1                 RSUM(100),       IWORK(13246)
      COMMON /PSDAT/   PSXY(2,505),  PSRAT(7,505),    PSTEMP(505),
     1                 PSFLOW(505),  PSHITE(505),     NPS
      EQUIVALENCE (ISOURC,WORK)
      DATA NVAR, NWPREC /17, 510/
      DEBUG = RNEG
      NWPRM2 = NWPREC - 2
      INDEX = IFIX(TIME + SMALL)/60
      IF(IWAS.LE.0) LINDEX = -1
      IF(INDEX.GT.LINDEX) GO TO 50
      IF(INDEX.EQ.LINDEX) RETURN
      KOK = -8888
      WRITE(6,1)
      RETURN
50    CONTINUE
CUNIVAC      READ(LUP,ERR=520,END=510) IDAT,NSTACK,(SUM(I),I=1,NWPRM2)
      BUFFER IN(LUP,1) (IDAT,SUM(NWPRM2))
      IF(UNIT(LUP))  60,510,520
60    IYR = IDAT/100000
      IDAY = IDAT/100 - IYR*1000
      IHR = IDAT - IYR*100000 - IDAY*100
      NWHR = NSTACK*NVAR
      NWLFT = NWHR - NWPRM2
      NBLKS = NWLFT/NWPREC
      IF(MOD(NWLFT,NWPREC).NE.0) NBLKS = NBLKS + 1
      IF(DEBUG.EQ.YES)  WRITE(6,2) IYR, IDAY, IHR, NSTACK
C     IF(DEBUG.EQ.YES) WRITE(6,11) (SUM(IJK),IJK=1,NWPRM2)
11    FORMAT(1H ,100(17F7.3,/,1X))
      IF(IHR.EQ.INDEX)  GO TO 100
      DO 80 I = 1,NBLKS
      READ(LUP)
80    CONTINUE
      GO TO 50
100   CONTINUE
      CALL XMIT(NWPRM2,SUM,WORK)
      K1 = -1
      K2 = NWPRM2
      DO 120 I = 1,NBLKS
      K1 = K1 + NWPREC
```

```
CUNIVAC  READ(LUP,ERR=520,END=510)  (SUM(J),J=1,NWPREC)
       BUFFER IN (LUP,1) (SUM(1), SUM(NWPREC))
110    IF(UNIT(LUP))       115,    510,    520
115    CALL XMIT(NWPREC,SUM,WORK(K1))
C      IF(I.EQ.1.AND.DEBUG.EQ.YES)WRITE(6,11)(SUM(IJK),IJK=1,NWPREC)
120    CONTINUE
C
       DO 150 K = 1,NSTACK
       N = (K-1)*NVAR
       PSXY(1,K) = WORK(1+N) - UTMXOR
       PSXY(2,K) = WORK(2+N) - UTMYOR
C        NOX,PARF,OLEF,AROM,ALDE,CO,SOX
       PSRAT(1,K) = WORK(9+N)*ADJUST(3)
       PSRAT(2,K) = WORK(13+N)*ADJUST(8)
       PSRAT(3,K) = WORK(14+N)*ADJUST(9)
       PSRAT(4,K) = WORK(15+N)*ADJUST(10)
       PSRAT(5,K) = WORK(16+N)*ADJUST(11)
       PSRAT(6,K) = WORK(11+N)*ADJUST(5)
       PSRAT(7,K) = WORK(8+N)*ADJUST(2)
       PSHITE(K) = FLOAT(ISOURC(3,K))
       PSTEMP(K) = FLOAT(ISOURC(5,K))
       PSFLOW(K) = WORK(6+N)
150    CONTINUE
C      DO 155 K= 1,5
C      WRITE(6,14) ITEMPR(K), IHITE(K)
C14     FORMAT(101X,2I10)
C      WRITE(6,12) PSXY(1,K),PSXY(2,K),(PSRAT(IS,K),IS=1,7),
C     1  PSFLOW(K), PSTEMP(K), PSHITE(K)
C12     FORMAT(1H ,12E10.3)
C155    CONTINUE
C
       NPS = NSTACK
       LINDEX = INDEX
       IF(NDSUMS.NE.IYES)    RETURN
C
C        REGIONAL SUMS
C
C        (NOTE
       NINM1 = NFLXIN - 1
       CALL XMIT(-NINM1,0,0,RSUM)
       DO 170 K = 1,NPS
       N = (K-1)*NVAR + 6
       DO 160 J = 1,NINM1
       RSUM(J) = RSUM(J) + WORK(J+N)
160    CONTINUE
170    CONTINUE
C
       DO 180 J = 1,NINM1
       N = J
       IF(J.GE.6)  N = N + 1
       RSUM(J) = RSUM(J)*ADJUST(N)
180    CONTINUE
C
       IF(IWAS.GT.0) GO TO 190
       CALL NEWPAG(TITLE,0,JDATE)
```

```
      WRITE(6,3) (NAMIN(I),I=1,5),(NAMIN(I),I=7,11)
190   WRITE(6,4) TIME,  (RSUM(I),I=1,NINM1)
      RETURN
510   KOK = 8
      WRITE(6,5)
      RETURN
520   KOK = 88
      WRITE(6,6)
      RETURN
C
1     FORMAT(48H0TRAJECTORY NOT CHRONOLOGICAL  --  JOB ABORTED    )
2     FORMAT(12H0    YEAR =  ,I6,7H  DAY = ,I4,8H  HOUR = ,I4,
     1  25H  NO. OF POINT SOURCES =  ,I5)
3     FORMAT(1H0,35HREGIONAL POINT SOURCE EMISSION SUMS  ,
     1  30X,12H(MOLES/HOUR)  ,//,8H   TIME  ,6X,11(A4,7X))
4     FORMAT(1H0,F7.1,2X,11E11.3)
5     FORMAT(50H0UNEXPECTED EOF ENCOUNTERED IN POINT SOURCE FILE
     1      ,30H      --          JOB ABORTED )
6     FORMAT(50H0PARITY ERROR IN POINT SOURCE FILE - JOB ABORTED  )
      END


      FUNCTION TRPLOG(X,XS,FS,NXF,NF)
C
C        LOG-LOG INTERPOLATOR FOR TABULAR FUNCTIONS
C        (ASSUMES LINEAR EXTENSIONS OF END SEGMENTS ON LOG-LOG
C        GRAPH TO CALCULATE BEYOND RANGE OF XS.)
C
      DIMENSION XS(NXF),  FS(NF,NXF)
C
      IF(X.GT.0.0)   GO TO 80
      WRITE(6,1) X
1     FORMAT(1H0,10HTRPLOG X=   ,E12.4)
      STOP
80    CONTINUE
      XLOG = ALOG(X)
      NXFM1 = NXF - 1
      IF( XS(NXF) - XS(1) )   100,  120,  120
100   DO 110 K = 2,NXFM1
      IF(X - XS(K))   110,  110,  140
110   CONTINUE
      K = NXF
      GO TO 140
120   DO 130 K = 2,NXFM1
      IF(XS(K) - X)  130,  130,  140
130   CONTINUE
      K = NXF
140   KL = K - 1
C
      FA = ALOG(FS(1,K))
      FB = ALOG(FS(1,KL))
      XA = ALOG(XS(K))
      XB = ALOG(XS(KL))
      FLOG = FB + (XLOG - XB)*(FA - FB)/(XA - XB)
      TRPLUG = EXP(FLOG)
      RETURN
      END



      BLOCK DATA
C
C         TABULAR DATA FOR HORIZONTAL(Y) AND VERTICAL(Z) SIGMAS AS
C         FUNCTIONS OF DOWNWIND DISTANCE FROM TURNERS WORKBOOK.
C
C         AND TALL STACK SIGMAS FROM ASME.
C
C
      COMMON /SIGMAS/   NDY,  XDY(4),  SIGYD(4,6),
     1                  NDZ,  XDZ(12), SIGZD(12,6),  NCAT
C
      COMMON /SIGTAL/ ATALL(6),BTALL(6),CTALL(6),DTALL(6)
C         HORIZONTAL IN KILOMETERS
C         VERTICAL IN METERS
C
      DATA NDY,  NDZ,  NCAT  / 4, 12, 6/
      DATA XDY /.10,  1.00,  10.00,  80.00/
      DATA SIGYD /
     A          .0270,   .212,   1.570,   9.000,
     B          .0190,   .157,   1.190,   6.800,
     C          .0125,   .104,    .840,   5.050,
     D          .0080,   .068,    .555,   3.350,
     E          .0060,   .050,    .410,   2.510,
     F          .0040,   .034,    .273,   1.680/
C
      DATA XDZ /  .100,.200,.300,.500,1.00,2.00,3.00,5.00,
     1          10.0,20.0,50.0,100. /
      DATA SIGZD/
     A         14.,29.5,48.0,105.,450.,1950.,4600.,13562.,
     A         58818.,255084.,1774117.,7694071.,
     B         10.8,20.3,30.2,51.0,110.,232.,365.,640.,
     B         1350.,2900.,7968.,17117.,
     C         7.4,13.9,20.1,32.0,61.,116.,169.,267.,
     C         500.,950.,2170.,4000.,
     D         4.6,8.5,12.1,18.6,31.5,50.0,64.5,89.0,
     D         137.,202.,328.,455.,
     E         3.5,6.38,8.8,13.0,21.3,33.7,42.7,56.0,
     E         79.0,110.,153.,183.,
     F         2.3,4.05,5.6,8.5,14.0,21.5,26.5,34.0,
     F         46.5,60.0,79.0,93.0/
C
C
      DATA ATALL /.40,.36,.32,.32,.31,.31/
      DATA BTALL /.91,.86,.78,.78,.71,.71/
      DATA CTALL /.40,.33,.22,.22,.06,.06/
      DATA DTALL /.91,.86,.78,.78,.71,.71/
C
      END
```

3. Chemical-Diffusion Module Listing

```
      PROGRAM KEMOD   (INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT,
     1                 TAPE1, TAPE3, TAPE4, TAPE10, TAPE11)
C
C     KEMOD IS THE DRIVER FOR THE  CHEMICAL/DIFFUSION MODULE.
C
C     SUBROUTINES REQUIRED
C             KEMOD2    DIFFUN    RATES     UPRAT2    UPFLX1    PREDAT
C             DIFCOF    SKEDUL    PEDERV    RATEMI    ISTATE    STEADY
C             JACOB     SOL282    UPFLX1    MATMUL    PHOTOD    SOLAR
C             TEMPR     XMIT      MDATE     SECOND    NEWPAG    MCHAR
C             SETPLT    COPLOT    FMAX      SCALE     TIMEX     UNMIXR
C             DRIVE     INTERP    TSTEP     COSET     PSET      BLKSOL
C             SOL       BLKDEC    DEC       ADJUST    CHECKY    UPSORC
C             ITHOUR    PRODUK
C
C     PERIPHERRALS REQUIRED
C             TAPE1 = PUNCHED OUTPUT OF SURFACE CONCENTRATIONS
C             TAPE3 = INTERMEDIATE I/O FOR PREDAT TO COPY INPUT
C             TAPE4 = SCRATCH FILE FOR PLOTTING ROUTINE
C             TAPE5 = INPUT
C             TAPE6 = OUTPUT
C             TAPE10 = VERIFICATION OF FLUX SCHEDULE UPDATES
C             TAPE11 = VERIFICATION OF KZ SCHEDULE UPDATES
C
      REAL INTIM
      COMMON/INPUTS/TITLE(20),           IDATE(10),          NCURV
      COMMON/FLUXES/FLXIN( 7,200),       FLXTIM(200),        NFLUX
      COMMON /PS1/  TPASS(200),          PS(7,5,75),         FRACT(3),
     1              NPTSR,               NPSFLX,             LOCPSF(7)
      COMMON/EPCOM6/PW(4500)
      DIMENSION  AF(11),   VFR(5,200),    PTSR(7,200)
      EQUIVALENCE (RNEG,NEG),(YES,IES),(RMOR,MORE)
      EQUIVALENCE (VFR,PW) , (PTSR,PW(1001))
      DATA YES,NEG,MORE,END/3HYES,2HNO,4HMORE,3HEND/
      DATA FRACT /.50,.50,0.0/
      DATA LIN,  LOUT  /3,  6/
      TERM = END
C
100   CALL PREDAT
C
      CALL MDATE(IDATE)
      READ(LIN,25) TITLE
      READ(LIN,29) GUAMED
      READ(LIN,29) FLXUNT
C
C        READ AREA SOURCE EMISSION FLUXES
C
      READ(LIN,24) NUMFLX
      DO 170 I=1,201
      READ(LIN,30)  A, (AF(N), N=1,NUMFLX)
      IF(A .LT. 0.0)  GO TO 180
      FLXTIM(I) = A
      DO 165 J=1,NUMFLX
      FLXIN(J,I) = AF(J)
165   CONTINUE

```

```
170   CONTINUE
      WRITE(LOUT,23)
      GO TO 600
180   NFLUX = I-1
      INTIM = FLXTIM(1)
      TSTOP = FLXTIM(NFLUX)
C
C          READ POINT SOURCE EMISSION FLUXES
C
      NPTSR = 0
      READ(LIN,24) NPSFLX
      IF(NPSFLX.EQ.0)  GO TO 500
      DO 400 K = 1,201
      READ(LIN,30) A, (AF(I),I=1,NPSFLX)
      IF(A.LT.0.0)  GO TO 450
      TPASS(K) = A
      DO 390 J = 1,NPSFLX
      PTSR(J,K) = AF(J)
390   CONTINUE
      READ(LIN,36)  (VFR(I,K), I=1,5)
400   CONTINUE
      WRITE(LOUT,37)
      GO TO 600
450   NPTSR = K - 1
C
500   CONTINUE
C
      IF (GOAHED.EQ.RNEG) GO TO 600
C
      CALL KEMOD2 (INTIM,TSTOP,NUMFLX,FLXUNT)
C
600   READ(LIN,25) TERM
      IF (TERM.EQ.RMOR) GO TO 100
      IF (TERM.NE.END) GO TO 600
      STOP
23    FORMAT (1H1,10X,37HTOO MANY AREA SOURCE FLUXES INPUT.
     1  23H  ---      JOB ABORTED.        )
24    FORMAT(40X,I10)
25    FORMAT (20A4)
29    FORMAT (40X,A4)
30    FORMAT (20X,F10.2,4E12.4,/,30X,4E12.4)
36    FORMAT (18X,5E12.4)
37    FORMAT (50H1,TOO MANY POINT SOURCES INPUT  --  JOB ABORTED     )
      END



      SUBROUTINE ADJUST (Y, NO)
C-----------------------------------------------------------------------
C THIS SUBROUTINE ADJUSTS THE Y ARRAY ON REDUCTION OF ORDER.
C SEE REFERENCE 1 FOR DETAILS.
C-----------------------------------------------------------------------
      INTEGER NO
      INTEGER JSTART, KFLAG, L, LMAX, METH, MF, N, NQ, NQINDX
      INTEGER I, IBACK, J, JP1, NQM1, NQM2
      DIMENSION Y(NO, 6)
C
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART
      COMMON /EPCM10/ TAU(13),EL(13),TQ(5),LMAX,METH,NQ,L,NQINDX
C(1
      DATA ONE /1.0D0/, ZERO /0.0D0/
      IF (NQ .EQ. 2) RETURN
      NQM1 = NQ - 1
      NQM2 = NQ - 2
      GO TO (100, 200), METH
C
 100  DO 110 J = 1,LMAX
 110    EL(J) = ZERO
      EL(2) = ONE
      HSUM = ZERO
      DO 130 J = 1,NQM2
C CONSTRUCT COEFFICIENTS OF X*(X+XI(1))*...*(X+XI(J)). ----------------
        HSUM = HSUM + TAU(J)
        XI = HSUM/H
        JP1 = J + 1
        DO 120 IBACK = 1,JP1
          I = (J + 3) - IBACK
 120      EL(I) = EL(I)*XI + EL(I-1)
 130    CONTINUE
C CONSTRUCT COEFFICIENTS OF INTEGRATED POLYNOMIAL. ---------------------
      DO 140 J = 2,NQM1
C(1
  140   EL(J+1) =  FLOAT(NQ)*EL(J)/ FLOAT(J)
      GO TO 300
C
 200  DO 210 J = 1,LMAX
 210    EL(J) = ZERO
      EL(3) = ONE
      HSUM = ZERO
      DO 230 J = 1,NQM2
C CONSTRUCT COEFFICIENTS OF X*X*(X+XI(1))*...*(X+XI(J)). --------------
        HSUM = HSUM + TAU(J)
        XI = HSUM/H
        JP1 = J + 1
        DO 220 IBACK = 1,JP1
          I = (J + 4) - IBACK
 220      EL(I) = EL(I)*XI + EL(I-1)
 230    CONTINUE
C
C SUBTRACT CORRECTION TERMS FROM Y ARRAY. -----------------------------
 300  DO 320 J = 3,NQ
        DO 310 I = 1,N
 310      Y(I,J) = Y(I,J) - Y(I,L)*EL(J)
 320    CONTINUE
      RETURN
C----------------------- END OF SUBROUTINE ADJUST ---------------------
      END
      SUBROUTINE BLKDEC (QB,NN,KK,IP,A,C,TL,W,NOGO,BCFLAG)
C
C     PERFORMS LU DECOMPOSITION ON THE MATRIX Q
C     WHERE Q IS BLOCK TRIDIAGONAL AS FOLLOWS
C             B(1)  C(1)  0 ........... 0
C             A(1)  B(2)  C(2) 0........ 0
C     Q =     0     A(2)  B(3)  C(3) 0..0
C                         .
C                          .
C                           .
C                  A(KK-2)  B(KK-1)  C(KK-1)
C                           A(KK-1)  B(KK)
C
C     THE MATRICES A,B,C ARE SQUARE MATRICES OF ORDER NN AND Q IS OF
C     BLOCK-ORDER KK.
C     IN OUR PROBLEM, THE MATRICES A(I) AND C(I) ARE SCALED IDENTITY
C     MATRICES, HENCE WE ONLY STORE ONE SCALE FACTOR PER MATRIX,
C     THUS A AND C ARE VECTORS OF LENGTH (KK-1).
C
C     IP IS A 2-DIMENSIONAL ARRAY OF ORDER (NN*KK) USED FOR STORING PIVO
C     INFORMATION IN THE LU DECOMPOSITION OF THE DIAGONAL SUBMATRICES B(
C
C     IN PRACTICE, THE MATRIX QB IS A 3-DIMENSIONAL ARRAY WHICH CONTAINS
C     ONLY THE DIAGONAL B SUBMATRICES.
C
C     TL IS A 3-DIMENSIONAL ARRAY USED USED TO STORE THE LOWER TRIANGULA
C     ELEMENTS OF Q.
C
C     W IS A WORK VECTOR OF LENGTH NN.
C
C     BCFLAG IS A BOUNDARY CONDITION CODE VECTOR - SEE KEMOD2 FOR
C     A COMPLETE DISCRIPTION.
C
C     NOGO = 0 IF SOLUTION IS SUCCESSFUL.
C     NOGO = 1 IF SINGULAR MATRIX IS ENCOUNTERED.
C
      DIMENSION     QB(NN,NN,KK),A(1),C(1),IP(NN,KK),TL(NN,NN,1),W(1)
      INTEGER BCFLAG(1)
C
C     BEGIN LU DECOMPOSITION OF MATRIX Q.  THE UPPER TRIANGULAR ELEMENTS
C     U, WILL BE STORED IN QB. THE LOWER TRIANGULAR ELEMENTS, L, WILL BE
C     STORED IN TL.
C
C     FIND L(I-1) FROM L(I-1)*U(I-1) = A(I-1) AND U(I) FROM
C     U(I) = B(I) - L(I-1)*C(I-1).
C     TO START, WE HAVE U(1) = B(1).
C
C     CHECK BOUNDARY CONDION CODE (BCFLAG)
      DO 100 I=1,NN
      IF(BCFLAG(I).NE.2)  GO TO 100
C     ADJUST B(1) FOR CONSTAT CONCENTRATION SURFACE BOUNDARY CONDITION
      DO 90 J=1,NN
      QB(I,J,1) = 0.0
   90 CONTINUE
      QB(I,I,1) =1.0
```

```
  100 CONTINUE
C
      NOGO = 0
      DO 300 I = 2,KK
       IM1 = I - 1
       CALL DEC(NN,NN,QB(1,1,IM1),IP(1,IM1),IER)
       IF(IER.NE.0) GO TO 420
C
C     FIND L(I-1). INVERT U(I-1) BY REPEATED CALLS TO SOL.
C
       SF = A(IM1)
        DO 200 J = 1,NN
         CALL XMIT(-NN,0.,W)
         W(J) = 1.0
         CALL SOL(NN,NN,QB(1,1,IM1),W,IP(1,IM1))
          DO 190 K = 1,NN
           TL(K,J,IM1) = SF*W(K)
  190     CONTINUE
  200   CONTINUE
C
C     FIND U(I).
C
       SF = C(IM1)
      IF(IM1.NE.1)  GO TO 240
C
      DO 220 J=1,NN
C     DECOUPLE  EQUATIONS FOR CONSTANT CONCENTRATION BCFLAG
      IF(BCFLAG(J).EQ.2)  GO TO 220
      DO 210 K=1,NN
      QB(K,J,I) = QB(K,J,I) -SF*TL(K,J,IM1)
  210 CONTINUE
  220 CONTINUE
      GO TO 300
C
  240 CONTINUE
        DO 250 J = 1,NN
        DO 250 K = 1,NN
         QB(K,J,I) = QB(K,J,I) - SF*TL(K,J,IM1)
  250   CONTINUE
  300 CONTINUE
C
C     PERFORM LU DECOMPOSITION ON U(KK).  THIS IS REQUIRED FOR THE
C     BACK SUBSTITUTION LATER ON.
C
      CALL DEC(NN,NN,QB(1,1,KK),IP(1,KK),IER)
      IM1 = KK
      IF(IER.NE.0) GO TO 420
      RETURN
C
C     MATRIX IS SINGULAR.  JOB IS ABORTED.
C
 420  CONTINUE
      NOGO = 1
      WRITE(6,1)  IM1
    1 FORMAT(1H0,5X, *SINGULAR MATRIX NO. *,I2,*.  JOB ABORTED.*)
```

```
      RETURN
      END


      SUBROUTINE BLKSOL(QB,NN,KK,RHS,IP,A,C,TL,W,BCFLAG)
C
C     SOLVES A SYSTEM OF LINEAR EQUATIONS Q*X = RHS AFTER
C     LU DECOMPOSITION HAS BEEN PERFORMED ON Q AND
C     WHERE Q IS BLOCK TRIDIAGONAL AS FOLLOWS
C             B(1)  C(1)  0 ........... 0
C             A(1)  B(2)  C(2) 0....... 0
C     Q =     0     A(2)  B(3)  C(3) 0..0
C                         .
C                          .
C                           .
C                    A(KK-2)  B(KK-1)  C(KK-1)
C                             A(KK-1)  B(KK)
C
C     THE MATRICES A,B,C ARE SQUARE MATRICES OF ORDER NN AND Q IS OF
C     BLOCK-ORDER KK.  RHS IS THE RIGHT-HAND SIDE OF THE EQUATION AND IS
C     A VECTOR OF LENGTH (NN*KK).  THE SOLUTION IS RETURNED IN RHS.
C     IN OUR PROBLEM, THE MATRICES A(I) AND C(I) ARE SCALED IDENTITY
C     MATRICES, HENCE WE ONLY STORE ONE SCALE FACTOR PER MATRIX,
C     THUS A AND C ARE VECTORS OF LENGTH (KK-1).
C
C     IP IS A 2-DIMENSIONAL ARRAY OF ORDER (NN*KK) USED FOR STORING PIVO
C     INFORMATION IN THE LU DECOMPOSITION OF THE DIAGONAL SUBMATRICES B(
C
C     IN PRACTICE, THE MATRIX QB IS A 3-DIMENSIONAL ARRAY WHICH CONTAINS
C     THE DIAGONAL B SUBMATRICES.
C
C     TL IS A 3-DIMENSIONAL ARRAY USED USED TO STORE THE LOWER TRIANGULA
C     ELEMENTS OF Q.
C
C     W IS A WORK VECTOR OF LENGTH NN.
C
C
C     BCFLAG IS A BOUNDARY CONDITION CODE VECTOR - SEE KEMOD2 FOR
C     A COMPLETE DISCRIPTION.
C
      DIMENSION     QB(NN,NN,KK),A(1),C(1),IP(NN,KK),RHS(NN,KK),
     1              TL(NN,NN,1),W(1)
      INTEGER BCFLAG(1)
C
C
C     FRONT SUBSTITUTION.
C     INITIALLY, Y(1) = RHS(1). THE RHS VECTOR IS DESTROYED IN THIS
C     OPERATION.
C
      DO 350 I = 2,KK
       IM1 = I - 1
       CALL MATMUL(TL(1,1,IM1),RHS(1,IM1),W,NN,NN,1)
        DO 340 J = 1,NN
         RHS(J,I) = RHS(J,I) - W(J)
```

```
  340   CONTINUE
  350 CONTINUE
C
C     BACK SUBSTITUTION.
C     THE VECTOR RHS PREVIOUSLY COMPUTED IN LOOP 350 IS ONCE AGAIN THE
C     RIGHT-HAND SIDE.  THE SOLUTION VECTOR IS AGAIN STORED IN RHS.
C     RECALL THAT THE U MATRICES IN QB HAVE ALREADY UNDERGONE LU DECOMPO
C     AND THE PIVOT INFORMATION IS STORED IN IP.
C
      CALL SOL(NN,NN,QB(1,1,KK),RHS(1,KK),IP(1,KK))
      N = KK - 2
      DO 400 I = 1,N
       K = KK - I
       J = K + 1
       SF = C(K)
        DO 390 M = 1,NN
         RHS(M,K) = RHS(M,K) - SF*RHS(M,J)
  390   CONTINUE
       CALL SOL(NN,NN,QB(1,1,K),RHS(1,K),IP(1,K))
  400 CONTINUE
C
C     SOLVE FOR RHS(1,1) WITH BOUNDARY CONDITION CHECK
      SF = C(1)
      DO 410 M = 1,NN
C     DECOUPLE EQUATIONS WITH CONSTANT CONCENTRATION BOUNDARY CONDITION
      IF(BCFLAG(M).EQ.2)  GO TO 410
      RHS(M,1) = RHS(M,1) - SF*RHS(M,2)
  410 CONTINUE
      CALL SOL(NN,NN,QB(1,1,1),RHS(1,1),IP(1,1) )
       RETURN
C
      END



      SUBROUTINE CHECKY(T,Y,YO,N,KOK)
C
C       CHECKY CHECKS FOR NEGATIVE CONCENTRATIONS
C       AND DEPLETION OF NITRIC OXIDE.
C
      DIMENSION INO(5),  Y(1),  YO(1)
      COMMON/CHEM1/ NOSTAT,    NOSTM1,    NOREAC
     1                NOSPEC,    NSTEDY,    NK
      DATA YMIN, YNOMIN,  KMAX  /1.E-18,1.E-6, 5/
      KNTER = 0
      KSPEC = 1
C         CHECK FOR NITRIC OXIDE DEPLETION
      DO 10 K = 1, NOSTAT
      KS = (K - 1)* NK + 1
      IF(Y(KS) .GT. YNOMIN)  GO TO 10
      KNTER = KNTER + 1
      Y(KS) = YO(KS)
C     WRITE(8,1) T, KSPEC, Y(KS), YO(KS)
10    CONTINUE
C         CHECK FOR Y VALUES LESS THAN YMIN
```

```
      DO 30 K = 1,N
      IF(Y(K).GT.YMIN)  GO TO 30
      KNTER = KNTER + 1
      KSPEC = MOD(K,NK)
C     WRITE(8,1) T,KSPEC, Y(K), YO(K)
      Y(K) = AMAX1(Y(K),YO(K))
30    CONTINUE
      IF(KNTER .GT. 5)  KOK = -100
      RETURN
1     FORMAT(1H 5X,6HTIME =,F8.2,13H   SPEC NO. =   ,I3,3X,
     1 20HOLD CONCENTRATION =,E13.4,3X,22HRESET CONCENTRATION =,E13.4)
      END


      SUBROUTINE COPLOT(NDPT)
C
C         COPLOT GENERATES A PRINTER-PLOT OF GROUND CONCENTRATIONS
C         VERSUS TIME FOR UP TO FIVE SPECIES
C
      COMMON/PRPLOT/     KPS( 5),          KSYM( 5),        GRCON(5,100),
     1                   TOUT(100),        VALMAX(5)
      COMMON /INPUTS/  TITLE(20),          IDATE(10),       NCURV
      DIMENSION KNAM(5)
      DATA KNAM /1HN,1H2,1HO,1HS,1H4/
      DATA ZPLUS/1.E-30/
      DATA LOUT /6/
C
      GO TO 30
C
      ENTRY IPLOT
      CALL XMIT(NCURV,KNAM,KSYM)
      CALL XMIT(-NCURV,0.,VALMAX)
      RETURN
C
   30 CONTINUE
C     T1 = TOUT(1) + ZPLUS
C     CALL SCALE(T1,TOUT(NDPT),XO,XR)
C     -----  FIX HORIZONTAL SCALE FOR ALL DAY  -----
      IF(TOUT(NDPT)-TOUT(1) .GT. 750.)  GO TO 35
      XO = TOUT(1)
      XR = XO + 750.
   35 CONTINUE
      CALL FMAX(VALMAX,NCURV,A)
      CALL SCALE(ZPLUS,A,YB,YT)
      IF(2.*A.LT.YT)  YT = .5*YT
      CALL SETPLT(XO,XR,YB,YT)
      DO 50 I=1,NDPT
      DO 40 K=1,NCURV
      CALL PLTPNT(TOUT(I),GRCON(K,I),KSYM(K))
   40 CONTINUE
   50 CONTINUE
      CALL PLTOUT(LOUT)
      WRITE(LOUT,1000)
      WRITE(LOUT,1001) TITLE
      RETURN
C
1000  FORMAT(1H ,60X,16HTIME  (MINUTES)        )
1001  FORMAT(1H0,39X,40HLEGEND
     1   20H S = S02   C = .1*CO        //,30X,20A4  )
      END


      SUBROUTINE COSET
C-----------------------------------------------------------------------
C COSET IS CALLED BY TSTEP AND SETS COEFFICIENTS FOR USE THERE.
C-----------------------------------------------------------------------
      INTEGER JSTART, KFLAG, L, LMAX, METH, MF, N, NQ, NQINDX
      INTEGER I, IBACK, J, JP1, JSTART, KFLAG, L, LMAX, MAXDER,
     1        METH, MF,N, NQ, NQINDX, NQM1
      DIMENSION EM(13)
C
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART
      COMMON /EPCM10/ TAU(13),EL(13),TQ(5),LMAX,METH,NQ,L,NQINDX
C(2
      DATA CORTES /0.100/
      DATA ONE  /1.0E0/, SIX /6.0E0/, TWO /2.0E0/, ZERO /0.0E0/
      AHOSS = ONE
C(2
      IF (SS .NE. ZERO) AHOSS =  ABS(H)/SS
      FLOTL =  FLOAT(L)
      NQM1 = NQ - 1
      GO TO (1, 2), METH
 1    MAXDER = 12
      GO TO 100
C
 2    MAXDER = 5
      GO TO 200
C
 100  IF (NQ .NE. 1) GO TO 110
      EL(1) = ONE
      EL(2) = ONE
      TQ(1) = ONE
      TQ(2) = TWO*AHOSS
      TQ(3) = SIX*TQ(2)
      TQ(5) = ONE
      GO TO 300
 110  HSUM = H
      EM(1) = ONE
      FLOTNQ = FLOTL - ONE
      DO 115 I = 2,L
 115    EM(I) = ZERO
      DO 150 J = 1,NQM1
        IF ((J .NE. NQM1) .OR. (NQINDX .NE. 1)) GO TO 130
        S = ONE
        CSUM = ZERO
        DO 120 I = 1,NQM1
C(1
          CSUM = CSUM + S*EM(I)/ FLOAT(I+1)
 120      S = -S
        TQ(1) = AHDSS*EM(NQM1)/(FLOTNQ*CSUM)
 130    RXI = H/HSUM
        DO 140 IBACK = 1,J
          I = (J + 2) - IBACK
 140      EM(I) = EM(I) + EM(I-1)*RXI
 150    HSUM = HSUM + TAU(J)
C COMPUTE INTEGRAL FROM -1 TO 0 OF POLYNOMIAL AND OF X TIMES IT. -------
      S = ONE
      EMO = ZERO
      CSUM = ZERO
      DO 160 I = 1,NQ
C(1
        FLOTI =  FLOAT(I)
        EMO = EMO + S*EM(I)/FLOTI
        CSUM = CSUM + S*EM(I)/(FLOTI+1)
 160    S = -S
C IN EL, FORM COEFFICIENTS OF NORMALIZED INTEGRATED POLYNOMIAL. --------
      S = ONE/EMO
      EL(1) = ONE
      DO 170 I = 1,NQ
C(1
  170   EL(I+1) = S*EM(I)/ FLOAT(I)
      XI = HSUM/H
      TQ(2) = AHDSS*XI*EMO/CSUM
      TQ(5) = XI/EL(L)
      IF (NQINDX .NE. 1) GO TO 300
C FOR HIGHER ORDER CONTROL CONSTANT, MULTIPLY POLYNOMIAL BY 1+X/XI(Q). -
      RXI = ONE/XI
      DO 180 IBACK = 1,NQ
        I = (L + 1) - IBACK
 180    EM(I) = EM(I) + EM(I-1)*RXI
C COMPUTE INTEGRAL OF POLYNOMIAL. -------------------------------------
      S = ONE
      CSUM = ZERO
      DO 190 I = 1,L
C(1
      CSUM = CSUM + S*EM(I)/ FLOAT(I+1)
 190    S = -S
      TQ(3) = AHDSS*FLOTL*EMO/CSUM
      GO TO 300
C
 200  DO 210 I = 3,L
 210    EL(I) = ZERO
      EL(1) = ONE
      EL(2) = ONE
      HSUM = H
      HSUM1 = ZERO
      PROD = ONE
      RXI = ONE
      IF (NQ .EQ. 1) GO TO 240
      DO 230 J = 1,NQM1
C IN EL, CONSTRUCT COEFFICIENTS OF (1+X/XI(1))*...*(1+X/XI(J+1)). ------
        HSUM = HSUM + TAU(J)
        HSUM1 = HSUM1 + TAU(J)
```

```
         PROD = PROD*(HSUM/HSUM1)
         RXI = H/HSUM
         JP1 = J + 1
         DO 220 IBACK = 1,JP1
           I = (J + 3) - IBACK
 220       EL(I) = EL(I) + EL(I-1)*RXI
 230     CONTINUE
 240  TQ(2) = AHDSS*EL(2)*(ONE + PROD)
      TQ(5) = (ONE + PROD)/EL(L)
      IF (NQINDX .NE. 1) GO TO 300
      CNQM1 = RXI/EL(L)
      ELP = EL(2) - RXI
      TQ(1) = AHDSS*ELP/CNQM1
      HSUM = HSUM + TAU(NQ)
      RXI = H/HSUM
      ELP = EL(2) + RXI
      TQ(3) = AHDSS*ELP*RXI*(ONE + PROD)*(FLOTL + ONE)
 300  TQ(4) = CORTES*TQ(2)
      LMAX = MAXDER + 1
      RETURN
C----------------------- END OF SUBROUTINE COSET ----------------------
      END


      SUBROUTINE DEC (N, NDIM, A, IP, IER)
C----------------------------------------------------------------------
C MATRIX TRIANGULARIZATION BY GAUSSIAN ELIMINATION.
C INPUT..
C    N = ORDER OF MATRIX.
C    NDIM = DECLARED DIMENSION OF ARRAY  A .
C    A = MATRIX TO BE TRIANGULARIZED.
C OUTPUT..
C    A(I,J), I.LE.J = UPPER TRIANGULAR FACTOR, U .
C    A(I,J), I.GT.J = MULTIPLIERS = LOWER TRIANGULAR FACTOR, I - L.
C    IP(K), K.LT.N = INDEX OF K-TH PIVOT ROW.
C    IP(N) = (-1)**(NUMBER OF INTERCHANGES) OR 0 .
C    IER = 0 IF A NONSINGULAR, OR K IF A FOUND TO BE
C          SINGULAR AT STAGE K.
C USE  SOL  TO OBTAIN SOLUTION OF LINEAR SYSTEM.
C DETERM(A) = IP(N)*A(1,1)*A(2,2)*...*A(N,N).
C IF IP(N)=0, A IS SINGULAR, SOL WILL DIVIDE BY ZERO.
C INTERCHANGES FINISHED IN U , ONLY PARTLY IN L .
C
C REFERENCE..
C C. B. MOLER, ALGORITHM 423, LINEAR EQUATION SOLVER,
C COMM. ASSOC. COMPUT. MACH., 15 (1972), P. 274.
C----------------------------------------------------------------------
      INTEGER IER, IP, N, NDIM
      INTEGER I, J, K, KP1, M, NM1
      DIMENSION A(NDIM,N),IP(N)
C(1
      DATA ONE /1.0D0/, ZERO /0.0D0/
      IER = 0
      IP(N) = 1
      IF (N .EQ. 1) GO TO 70                                            KM
      NM1 = N - 1                                                       KM
      DO 60 K = 1,NM1                                                   KM
        KP1 = K + 1                                                     KM
        M = K                                                           KM
        DO 10 I = KP1,N                                                 KM
 10       IF ( ABS(A(I,K)) .GT.  ABS(A(M,K))) M = I                     KM
        IP(K) = M                                                       KM
        T = A(M,K)                                                      KM
        IF (M .EQ. K) GO TO 20                                          KM
        IP(N) = -IP(N)                                                  KM
        A(M,K) = A(K,K)                                                 KM
        A(K,K) = T                                                      KM
 20     IF (T .EQ. ZERO) GO TO 80                                       KM
        T = ONE/T                                                       KM
        DO 30 I = KP1,N                                                 KM
 30       A(I,K) = -A(I,K)*T                                            KM
        DO 50 J = KP1,N                                                 KM
          T = A(M,J)                                                    KM
          A(M,J) = A(K,J)                                               KM
          A(K,J) = T                                                    KM
          IF (T .EQ. ZERO) GO TO 50                                     KM
          DO 40 I = KP1,N                                               KM
 40         A(I,J) = A(I,J) + A(I,K)*T                                  KM
 50       CONTINUE                                                      KM
 60     CONTINUE                                                        KM
 70   K = N                                                             KM
      IF (A(N,N) .EQ. ZERO) GO TO 80                                    KM
      RETURN                                                            KM
 80   IER = K                                                           KM
      IP(N) = 0                                                         KM
      RETURN                                                            KM
C-----------------------  END OF SUBROUTINE DEC  ----------------------KM
      END                                                               KM


      SUBROUTINE DIFCOF(NOSTAT)                                         KM
C                                                                       KM
C          THIS SUBROUTINE CALCULATES THE SCALED DIFFUSIVITY            KM
C          COEFFICIENTS FROM DFINIT                                     KM
C          VARIABLE MESH VERSION CODE 7.6.77                            KM
C                                                                       KM
      INTEGER BCFLAG                                                    KM
      COMMON/CHEM3/ ZEE(5),              DELZ(4),            HTCELL(6), KM
     1              TDELZ(2) ,           DFINIT(6),          SCALOW(4), KM
     2              DCOF(5),             FLXWAL(40),         FLXDGE(40),KM
     3              BCFLAG(40),          DPRATE(10),         DEPOWR(10),KM
     4              LOCDPF(10),          NDPFLX ,            SCALUP(4)  KM
C                                                                       KM
C          COEFFICIENTS FOR GROUND AND EDGE EQUATIONS                   KM
      I=1                                                               KM
      DCOF(I) =(DFINIT(I)+DFINIT(I+1))/HTCELL(I)/HTCELL(I)              KM
      SCALUP(I) = DCOF(I)                                               KM
      I=NOSTAT                                                          KM
      DCOF(I) =(DFINIT(I)+DFINIT(I+1))/HTCELL(I)/HTCELL(I)
      SCALOW(I-1) = DCOF(I)
C
C          COEFFICIENTS FOR INTERMIDIATE EQUATIONS
C
      NOSTM1 = NOSTAT - 1
      DO 100 I=2,NOSTM1
      SCALOW(I-1) = DFINIT(I)/DELZ(I-1)/HTCELL(I)
      SCALUP(I) = DFINIT(I+1)/DELZ(I)/HTCELL(I)
      DCOF(I) = SCALOW(I-1) + SCALUP(I)
  100 CONTINUE
      RETURN
      END



      SUBROUTINE DIFFUN (N,T,Y,YDOT)
C
C     ------       DIFFUN DEFINES THE DIFFERENTIAL EQUATIONS  ----------
C
C                      YDOT = F(Y,T)
C
      INTEGER BCFLAG
      COMMON/CHEM1/ NUSTAT,             NOSTM1,             NOREAC,
     1              NOSPEC,             NSTDY,              NK
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),         RATKON(55),
     1              RATEFF(55),         RATEV(2,5),         QRATE,
     2              NVRATE,             LOCVRT(2)
      COMMON/CHEM3/ ZEE(5),             DELZ(4),            HTCELL(6),
     1              TOELZ(2) ,          DFINIT(6),          SCALOW(4),
     2              DCOF(5),            FLXWAL(40),         FLXDGE(40),
     3              BCFLAG(40),         DPRATE(10),         DEPOWR(10),
     4              LOCDPF(10),         NDPFLX ,            SCALUP(4)
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT
      COMMON /CHEM5/  REACT(20,55),SPEC(40),LOCFLX(10),
     1                NASFLX,FLXW1(30),FLXW2(30)
      COMMON /PS1/   TPASS(200), PS(7,5,75), FRACT(3),
     1               NPTSR,NPSFLX,LOCPSF(7)
      DIMENSION  Y(N),     YDOT(N),     SINK(30)
C
      CALL XMIT (-NK, 0.0, SINK)
C
      CALL RATES(Y,YDOT)
C
      IF(NDPFLX.LT.1)  GO TO 50
C     ----    UPDATE FLUXES OF DEPOSITING SPECIES
      DO 40 K=1,NDPFLX
      ID = LOCDPF(K)
      IF(Y(ID).LT.1.E-16)  GO TO 40
      SINK(ID) = - DPRATE(K)*(Y(ID)**DEPOWR(K))
   40 CONTINUE
   50 CONTINUE
C
C        INTERPOLATE EMISSION FLUXES TO THE CURRENT TIME
C
```

```
      FDFLT = (T - TLAST)/UPDINT
      DO 60 I = 1,NASFLX
      K = LOCFLX(I)
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT
60    CONTINUE
      K = 7
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT
      K = 2
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT
      DO 75 J =1,NOSTAT
      DO 70 I =1,NPSFLX
      K =LOCPSF(I)
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT
70    CONTINUE
      K = 2
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT
      K = 7
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT
75    CONTINUE
C
      NM1 = NK*NOSTM1
      NM2 = NK*(NOSTAT-2)
C     ------- EQUATIONS FOR SURFACE AND TOP EDGE MESH POINTS
      DO 100  I=1,NK
      IL=NM1+I
      YDOT(I) = DCOF(1)*(Y(I+NK)-Y(I)) + YDOT(I)
     1                 +  TOELZ(1)*(FLXWAL(I)+SINK(I)) + PSRATE(I,1)
      IF(BCFLAG(I) .EQ. 2)  YDOT(I) = 0.0
      YDOT(IL) = DCOF(NOSTAT)*(Y(I+NM2)-Y(IL)) + YDOT(IL)
     1                              +  PSRATE(I,NOSTAT)
  100 CONTINUE
C     ------- EQUATIONS FOR INTERMEDIATE MESH POINTS
      DO 200 J=2,NOSTM1
      NM1= (J-2)*NK
      NN = (J-1)*NK
      NP1 = J*NK
      JM1 = J- 1
      DO 150 I = 1,NK
      LN = I + NN
      YDOT(LN) = SCALOW(JM1)*Y(I+NM1) - DCOF(J)*Y(LN) +
     1           SCALUP(J)*Y(I+NP1) + YDOT(LN) + PSRATE(I,J)
  150 CONTINUE
  200 CONTINUE
      RETURN
      END
      SUBROUTINE DRIVE(N,T0,H0,Y0,TOUT,EPS,IERROR,MF,INDEX,BIGSTP,KOK)
C
C-----------------------------------------------------------------------
C     THIS IS A SPECIAL VERSION OF EPISODE  (ERT CODE DATE 7.8.77)
C     FOR SOLVING A BLOCK TRI-DIAGONAL SYSTEM OF ODE*S FOR KEMOD2.
C     THIS VERSION OF EPISODE SHOULD ONLY BE USED WITH MF = 21
C     MODIFICATIONS BY F.W. LURMANN   (7.8.77).
C-----------------------------------------------------------------------
C
C THIS IS BASICALLY THE JUNE 24, 1975 VERSION OF
C EPISODE..  EXPERIMENTAL PACKAGE FOR INTEGRATION OF
C SYSTEMS OF ORDINARY DIFFERENTIAL EQUATIONS,
C    DY/DT = F(Y,T),  Y = (Y(1),Y(2),...,Y(N)) TRANSPOSE,
C GIVEN THE INITIAL VALUE OF Y.
C                 REFERENCES
C 1.  G. D. BYRNE AND A. C. HINDMARSH, A POLYALGORITHM FOR THE
C       NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS,
C       UCRL-75652, LAWRENCE LIVERMORE LABORATORY, P. O. BOX 808,
C       LIVERMORE, CA 94550, APRIL 1974. ALSO IN ACM TRANSACTIONS
C       ON MATHEMATICAL SOFTWARE, 1 (1975), PP. 71-96.
C
C 2.  A. C. HINDMARSH AND G. D. BYRNE, EPISODE.. AN EXPERIMENTAL
C       PACKAGE FOR THE INTEGRATION OF SYSTEMS OF ORDINARY
C       DIFFERENTIAL EQUATIONS, UCID-30112, L.L.L., MAY, 1975.
C
C 3.  A. C. HINDMARSH, GEAR.. ORDINARY DIFFERENTIAL EQUATION
C       SYSTEM SOLVER, UCID-30001, REV. 3, L.L.L., DECEMBER, 1974.
C
C-----------------------------------------------------------------------
C DRIVE IS A DRIVER SUBROUTINE FOR THE EPISODE PACKAGE.
C DRIVE IS TO BE CALLED ONCE FOR EACH OUTPUT VALUE OF T.
C IT THEN MAKES REPEATED CALLS TO THE CORE INTEGRATOR
C SUBROUTINE, TSTEP.
C
C THE INPUT PARAMETERS ARE AS FOLLOWS.
C   N     =  THE NUMBER OF DIFFERENTIAL EQUATIONS (USED ONLY ON
C              FIRST CALL, UNLESS INDEX = -1).  N MUST NEVER BE
C              INCREASED DURING A GIVEN PROBLEM.
C   T0    =  THE INITIAL VALUE OF T, THE INDEPENDENT VARIABLE
C              (USED FOR INPUT ONLY ON FIRST CALL).
C   H0    =  THE STEP SIZE IN T (USED FOR INPUT ONLY ON THE
C              FIRST CALL, UNLESS INDEX = 3 ON INPUT).  WHEN
C              INDEX = 3, H0 IS THE MAXIMUM ABSOLUTE VALUE OF
C              THE STEP SIZE TO BE USED.
C   Y0    =  A VECTOR OF LENGTH N CONTAINING THE INITIAL VALUES OF
C              Y (USED FOR INPUT ONLY ON FIRST CALL).
C   TOUT  =  THE VALUE OF T AT WHICH OUTPUT IS DESIRED NEXT.
C              INTEGRATION WILL NORMALLY GO BEYOND TOUT AND
C              INTERPOLATE TO T = TUUT.  (USED ONLY FOR INPUT.)
C   EPS   =  THE RELATIVE ERROR BOUND (USED ONLY ON FIRST CALL,
C              UNLESS INDEX = -1).  THIS BOUND IS USED AS FOLLOWS.
C              LET R(I) DENOTE THE ESTIMATED RELATIVE LOCAL ERROR
C              IN Y(I), I.E. THE ERROR RELATIVE TO YMAX(I), AS
C              MEASURED PER STEP (OF SIZE H) OR PER SS UNITS OF T.
C              THEN EPS IS A BOUND ON THE ROOT-MEAN-SQUARE NORM
```

```
C                   OF THE VECTOR R, I.E.
C                            N
C                   SQRT ( SUM ( R(I)**2 )/N ) .LT. EPS.
C                           I=1
C                   THE VECTOR YMAX IS COMPUTED IN DRIVE AS DESCRIBED
C                   UNDER IERROR BELOW.
C                   IF ERROR CONTROL PER SS UNITS OF T IS DESTRED, SET SS
C                   TO A POSITIVE NUMBER AFTER STATEMENT 10 (WHERE IT IS
C                   NOW SET TO ZERO), AND UPDATE IT AFTER STATEMENT 60.
C                   SEE ALSO THE COMMENTS ON SS AND YMAX BELOW.
C     IERROR =  THE ERROR FLAG WITH VALUES AND MEANINGS AS FOLLOW.
C            1      ABSOLUTE ERROR IS CONTROLLED.  YMAX(I) = 1.0.
C            2      ERROR RELATIVE TO ABS(Y) IS CONTROLLED.  IF Y(I) = 0.0
C                   A DIVIDE ERROR WILL OCCUR.  YMAX(I) = ABS(Y(I)).
C            3      ERROR RELATIVE TO THE LARGEST VALUE OF ABS(Y(I)) SEEN
C                   SO FAR IS CONTROLLED.  IF THE INITIAL VALUE OF Y(I) IS
C                   0.0, THEN YMAX(I) IS SET TO 1.0 INITIALLY AND REMAINS
C                   AT LEAST 1.0.
C      MF       =   THE METHOD FLAG (USED ONLY ON FIRST CALL, UNLESS
C                   INDEX = -1).  ALLOWED VALUES ARE 10, 11, 12, 13,
C                   20, 21, 22, 23.  MF IS AN INTEGER WITH TWO DECIMAL
C                   DIGITS, METH AND MITER (MF = 10*METH + MITER).  (MF
C                   CAN BE THOUGHT OF AS THE ORDERED PAIR (METH,MITER).)
C                   METH IS THE BASIC METHOD INDICATOR.
C                     METH = 1 INDICATES VARIABLE-STEP SIZE, VARIABLE-
C                              ORDER ADAMS METHOD, SUITABLE FOR NON-
C                              STIFF PROBLEMS.
C                     METH = 2 INDICATES VARIABLE-STEP SIZE, VARIABLE-
C                              ORDER BACKWARD DIFFERENTIATION METHOD,
C                              SUITABLE FOR STIFF PROBLEMS.
C                   MITER INDICATES THE METHOD OF ITERATIVE CORRECTION
C                     (NONLINEAR SYSTEM SOLUTION).
C                     MITER = 0 INDICATES FUNCTIONAL ITERATION (NO
C                               PARTIAL DERIVATIVES NEFDED).
C                     MITER = 1 INDICATES A CHORD OR SEMI-STATIONARY
C                               NEWTON METHOD WITH CLOSED FORM (EXACT)
C                               JACOBIAN, WHICH IS COMPUTED IN THE
C                               USER SUPPLIED SUBROUTINE
C                               FEDERV(N,T,Y,PD,NO) DESCRIBED BELOW.
C                     MITER = 2 INDICATES A CHORD  OR SEMI-STATIONARY
C                               NEWTON METHOD WITH AN INTERNALLY
C                               COMPUTED FINITE DIFFERENCE APPROXIMATION
C                               TO THE JACOBIAN.
C                     MITER = 3 INDICATES A CHORD OR SEMI-STATIONARY
C                               NEWTON METHOD WITH AN INTERNALLY
C                               COMPUTED DIAGONAL MATRIX APPROXIMATION
C                               TO THE JACOBIAN, BASED ON A DIRECTIONAL
C                               DERIVATIVE.
C      INDEX =  INTEGER USED ON INPUT TO INDICATE TYPE OF CALL,
C                  WITH THE FOLLOWING VALUES AND MEANINGS..
C            1     THIS IS THE FIRST CALL FOR THIS PROBLEM.
C            0     THIS IS NOT THE FIRST CALL FOR THIS PROBLEM,
C                  AND INTEGRATION IS TO CONTINUE.
C           -1     THIS IS NOT THE FIRST CALL FOR THE PROBLEM,
C                  AND THE USER HAS RESET N, EPS, AND/OR MF.
```

```
C        2    SAME AS 0 EXCEPT THAT TOUT IS TO BE HIT
C             EXACTLY (NO INTERPOLATION IS DONE).
C             ASSUMES TOUT .GE. THE CURRENT T.
C        3    SAME AS 0 EXCEPT CONTROL RETURNS TO CALLING
C             PROGRAM AFTER ONE STEP.  TOUT IS IGNORED.
C           SINCE THE NORMAL OUTPUT VALUE OF INDEX IS 0,
C           IT NEED NOT BE RESET FOR NORMAL CONTINUATION.
C
C     BIGSTP = MAXIMUM STEP SIZE ALLOWED    (J.R. MARTINEZ --  6.15.77)
C
C------------------------------------------------------------------
C
      INTEGER IERROR, INDEX, MF, N
      INTEGER IPIV, JSTART, KFLAG, MFC, NC, NFE, NJE,
     1        NQUSED, NSQ, NSTEP
      INTEGER I, KGO, NHCUT, NO
      INTEGER LOUT
C(6
C/14
      DIMENSION Y(150,6)
      DIMENSION Y0(N)
C
      COMMON /EPCOM1/T,H,HMIN,HMAX,EPSC,SS,UROUND,NC,MFC,KFLAG,JSTART
      COMMON /EPCOM2/ YMAX(150)
      COMMON /EPCOM3/ ERROR(150)
      COMMON /EPCOM4/ SAVE1(150)
      COMMON /EPCOM5/ SAVE2(150)
      COMMON /EPCOM6/ PW(4500)
      COMMON /EPCOM7/ IPIV(150)
      COMMON /EPCOM8/ EPSJ,NSQ
      COMMON /EPCOM9/ HUSED,NQUSED,NSTEP,NFE,NJE
      COMMON /EPCM12/ PWL(3600),  WORKR(30), DUPPER(4), DLOWER(4)
      INTEGER BCFLAG
      COMMON/CHEM1/ NOSTAT,               NOSTM1,              NOREAC,
     1              NOSPEC,               NSTDY,               NK
      COMMON/CHEM2/ CONIN(40,5),          WTMOLE(40),          RATKON(55),
     1              RATEFF(55),           RATEV(2,5),          ORATE,
     2              NVRATE,               LOCVRT(2)
      COMMON/CHEM3/ ZEE(5),               DELZ(4),             HTCELL(6),
     1              TDELZ(2) ,            DFINIT(6),           SCALOW(4),
     3              BCFLAG(40),           DPRATE(10),          DEPOWR(10),
     2              DCOF(5),              FLXWAL(40),          FLXDGE(40),
     4              LOCDPF(10),           NDPFLX ,             SCALUP(4)
C
      DATA LOUT /6/
C(3
      DATA HCUT /0.100/
      DATA FOUR /4.000/, HUNDRD /1.0E2/, ONE  /1.000/,
     1     TEN  /1.0E1/, ZERO   /0.000/
      DATA ERLIMT , KKKER  /1.0E-2,  0/
C)3
      IF (INDEX .EQ. 0) GO TO 20
      IF (INDEX .EQ. 2) GO TO 25
      IF (INDEX .EQ. -1) GO TO 30
      IF (INDEX .EQ. 3) GO TO 40
      IF (INDEX .NE. 1) GO TO 430
      IF (EPS .LE. ZERO) GO TO 400
      IF (N .LE. 0) GO TO 410
      IF (N .GT.150) GO TO 440
      IF ((T0-TOUT)*H0 .GE. ZERO) GO TO 420
C-----------------------------------------------------------------------
C IF INITIAL VALUES FOR YMAX OTHER THAN THOSE BELOW ARE DESIRED,
C THEY SHOULD BE SET HERE.  ALL YMAX(I) MUST BE POSITIVE.  IF
C VALUES FOR HMIN OR HMAX, THE BOUNDS ON THE ABSOLUTE VALUE OF H,
C OTHER THAN THOSE BELOW, ARE DESIRED, THEY ALSO SHOULD BE SET HERE.
C IF ERROR PER SS UNITS OF T IS TO BE CONTROLLED, SS SHOULD BE SET
C TO A POSITIVE VALUE BELOW.  ERROR PER UNIT STEP IS CONTROLLED
C WHEN SS = 1.  THE DEFAULT VALUE FOR SS IS 0 AND YIELDS CONTROL
C OF ERROR PER STEP.
C-----------------------------------------------------------------------
C SET UROUND, THE MACHINE ROUNDOFF CONSTANT, HERE.
C USE STATEMENT BELOW FOR SHORT PRECISION ON IBM 360 OR 370.
C     UROUND = 9.53674E-7
C USE STATEMENT BELOW FOR SINGLE PRECISION ON CDC 7600 OR 6600.
C     UROUND = 7.105427406E-15
C USE STATEMENT BELOW FOR SHORT PRECISION ON UNIVAC 1110.
C     UROUND = 1.490116112E-8
C-----------------------------------------------------------------------
      UROUND=7.105427406E-15
      DO 10 I = 1,N
        GO TO (5, 6, 7), IERROR
C IERROR   =   1, 2, 3  ---------------------------------------------------
  5     YMAX(I) = ONE
        GO TO 10
  6     YMAX(I) =  ABS(Y0(I))
        GO TO 10
  7     YMAX(I) =  ABS(Y0(I))
      YMAX(I) = AMAX1( YMAX(I) , ERLIMT)
        IF (YMAX(I) .EQ. ZERO) YMAX(I) = ONE
 10     Y(I,1) = Y0(I)
      NC = N
      T = T0
      H = H0
      IF ((T+H) .EQ. T) WRITE(LOUT,15) T
 15   FORMAT(/46H---  MESSAGE FROM SUBROUTINE DRIVE IN EPISODE,,
     1        24H THE O.D.E. SOLVER.  ---/22H WARNING.. T + H = T =,
     2        E18.8,18H IN THE NEXT STEP./)
C(2
      HMIN =  ABS(H0)
      HMAX =  ABS(T0 - TOUT)*TEN
      HMAX=BIGSTP
      EPSC = EPS
      MFC = MF
      JSTART = 0
      SS = ZERO
      N0 = N
      NSQ = N0*N0
C(1
      EPSJ =  SQRT(UROUND)
      NHCUT = 0
      GO TO 50
C TOP IS THE PREVIOUS OUTPUT VALUE OF TO FOR USE IN HMAX. --------------
C(3
   20 HMAX =  ABS(TOUT - TOP)*TEN
      HMAX=BIGSTP
      GO TO 80
   25 HMAX =  ABS(TOUT - TOP)*TEN
      HMAX=BIGSTP
C
       IF ((T-TOUT)*H .GE. ZERO) GO TO 460
       GO TO 45
C
 30   IF ((T-TOUT)*H .GE. ZERO) GO TO 450
      IF (MF .NE. MFC) JSTART = -1
      NC = N
      EPSC = EPS
      MFC = MF
      GO TO 45
C
 40   HMAX = H0
C
 45   IF ((T+H) .EQ. T) WRITE(LOUT,15) T
C
 50   CALL TSTEP (Y, N0)
C
      KGO = 1 - KFLAG
      GO TO (60, 100, 200, 300), KGO
C KFLAG  =   0,  -1,  -2,  -3  ---------------------------------------
C
 60   CONTINUE
C---------------------------------------------------------------------
C NORMAL RETURN FROM TSTEP.
C
C THE WEIGHTS YMAX(I) ARE UPDATED.  IF DIFFERENT VALUES ARE DESIRED,
C THEY SHOULD BE SET HERE.  IF SS IS TO BE UPDATED FOR CONTROL OF
C ERROR PER SS UNITS OF T, IT SHOULD ALSO BE DONE HERE.  A TEST IS
C MADE TO DETERMINE IF EPS IS TOO SMALL FOR MACHINE PRECISION.
C
C ANY OTHER TESTS OR CALCULATIONS THAT ARE REQUIRED AFTER EACH STEP
C SHOULD BE INSERTED HERE.
C
C IF INDEX = 3, Y0 IS SET TO THE CURRENT Y VALUES ON RETURN.
C IF INDEX = 2, H IS CONTROLLED TO HIT TOUT (WITHIN ROUNDOFF
C ERROR), AND THEN THE CURRENT Y VALUES ARE PUT IN Y0 ON
C RETURN.  FOR ANY OTHER VALUE OF INDEX, CONTROL RETURNS TO
C THE INTEGRATOR UNLESS TOUT HAS BEEN REACHED.  THEN
C INTERPOLATED VALUES OF Y ARE COMPUTED AND STORED IN Y0 ON
C RETURN.
C IF INTERPOLATION IS NOT DESIRED, THE CALL TO INTERP SHOULD
C BE DELETED AND CONTROL TRANSFERRED TO STATEMENT 500 INSTEAD
C OF 520.
C---------------------------------------------------------------------
C...........CHECK FOR NEGATIVE VALUES OF Y
      CALL CHECKY(T,Y,Y0,N,KOK)
      IF(KOK .LT. 0) RETURN
```

```
C...........COMPUTE STEADY STATE SPECIES
      CALL STEADY(Y,N)
      KKKER = KKKER + 1
      D = ZERO
      DO 70 I = 1,N
        AYI =  ABS(Y(I,1))
        GO TO (70, 66, 67), IERROR
C IERROR  =     1,  2,  3  ---------------------------------------------
 66     YMAX(I) = AYI
        GO TO 70
  67    YMAX(I) = AMAX1(YMAX(I), AYI)
      IF(KKKER.LT.20)  GO TO 70
C .........UPDATE YMAX EVERY 20 STEPS USING AYI AND ERLIMT
      KKKER = 0
      YMAX(I) = AMAX1(AYI,ERLIMT)
  70    D = D + (AYI/YMAX(I))**2
      D = D*(UROUND/EPS)**2
      IF (D .GT.  FLOAT(N)) GO TO 250
      IF (INDEX .EQ. 3) GO TO 500
      IF (INDEX .EQ. 2) GO TO 85
 80   IF ((T-TOUT)*H .LT. ZERO) GO TO 45
      CALL INTERP (TOUT, Y, NO, YO)
      TO = TOUT
      GO TO 520
 85   IF (((T+H)-TOUT)*H .LE. ZERO) GO TO 45
C(1
      IF ( ABS(T-TOUT) .LE. HUNDRD*UROUND*HMAX) GO TO 500
      IF ((T-TOUT)*H .GE. ZERO) GO TO 500
      H = (TOUT - T)*(ONE - FOUR*UROUND)
      JSTART = -1
      GO TO 45
C-----------------------------------------------------------------------
C ON AN ERROR RETURN FROM TSTEP, AN IMMEDIATE RETURN OCCURS IF
C KFLAG = -2, AND RECOVERY ATTEMPTS ARE MADE OTHERWISE.
C TO RECOVER, H AND HMIN ARE REDUCED BY A FACTOR OF .1 UP TO 10
C TIMES BEFORE GIVING UP.
C-----------------------------------------------------------------------
 100  WRITE (LOUT,101)
 101  FORMAT (/46H---  MESSAGE FROM SUBROUTINE DRIVE IN EPISODE,,
     1         24H THE O.D.E. SOLVER.  ---/)
      WRITE(LOUT,105) T,HMIN
 105  FORMAT(//35H KFLAG = -1 FROM INTEGRATOR AT T = ,E18.8/
     1        40H  ERROR TEST FAILED WITH ABS(H) = HMIN =,E18.8/)
 110  IF (NHCUT .EQ. 10) GO TO 150
      NHCUT = NHCUT + 1
      HMIN = HCUT*HMIN
      H = HCUT*H
      WRITE (LOUT,115) H
 115  FORMAT(24H  H HAS BEEN REDUCED TO ,E18.8,
     1        26H  AND STEP WILL BE RETRIED//)
      JSTART = -1
      GO TO 45
C
 150  WRITE (LOUT,155)
 155  FORMAT(//44H PROBLEM APPEARS UNSOLVABLE WITH GIVEN INPUT//)
```

```
      GO TO 500
C
  200 WRITE (LOUT,101)
      WRITE (LOUT,205) T,H,EPS
  205 FORMAT(//16H KFLAG = -2  T =,E18.8,4H H =,E18.8,6H EPS =,E18.8/
     1         50H  THE REQUESTED ERROR IS TOO SMALL FUR INTEGRATOR.//)
      GO TO 500
C
  250 WRITE (LOUT,101)
      WRITE (LOUT,255) T,EPS
  255 FORMAT(//46H INTEGRATION HALTED BY SUBROUTINE DRIVE AT T =,
     1         E18.8/43H EPS IS TOO SMALL FOR MACHINE PRECISION AND/
     2         29H PROBLEM BEING SOLVED.  EPS =,E18.8//)
      KFLAG = -2
      GO TO 500
C
  300 WRITE (LOUT,101)
      WRITE (LOUT,305) T
  305 FORMAT(//34H KFLAG = -3 FROM INTEGRATOR AT T =,E18.8/
     1         45H  CORRECTOR CONVERGENCE COULD NOT BE ACHIEVED/)
      GO TO 110
C
  400 WRITE (LOUT,101)
      WRITE (LOUT,405) EPS
  405 FORMAT(//35H ILLEGAL INPUT.. EPS .LE. 0. EPS = ,E18.8//)
      INDEX = -4
      RETURN
C
  410 WRITE (LOUT,101)
      WRITE (LOUT,415) N
  415 FORMAT(//31H ILLEGAL INPUT.. N .LE. 0. N = ,I8//)
      INDEX = -4
      RETURN
C
  420 WRITE (LOUT,101)
      WRITE (LOUT,425) TO,TOUT,HO
  425 FORMAT(//39H ILLEGAL INPUT.. (TO - TOUT)*HO .GE. 0./
     1         5H TO =,E18.8,7H TOUT =,E18.8,5H HO =,E18.8//)
      INDEX = -4
      RETURN
C
  430 WRITE (LOUT,101)
      WRITE (LOUT,435) INDEX
  435 FORMAT(//24H ILLEGAL INPUT.. INDEX =,I8//)
      INDEX = -4
      RETURN
C
  440 WRITE (LOUT,101)
      WRITE (LOUT,445) N
  445 FORMAT (//39H ILLEGAL INPUT.  THE NUMBER OF ORDINARY/
     1         43H DIFFERENTIAL EQUATIONS BEING SOLVED IS N =, I6/
     2         42H STORAGE ALLOCATION IN SUBROUTINE DRIVE IS/
     3         46H TOO SMALL.  SEE COMMENTS IN SUBROUTINE DRIVE./)
      INDEX = -4
      RETURN
```

```
C
 450  WRITE (LOUT,101)
      WRITE (LOUT,455) T,TOUT,H
 455  FORMAT(//46H INDEX = -1 ON INPUT WITH (T - TOUT)*H .GE. 0./
     1        44H INTERPOLATION WAS DONE AS ON NORMAL RETURN./
     2        41H DESIRED PARAMETER CHANGES WERE NOT MADE./
     3        4H T =,E18.8,7H TOUT =,E18.8,4H H =,E18.8//)
      CALL INTERP (TOUT, Y, NO, Y0)
      T0 = TOUT
      INDEX = -5
      RETURN
C
 460  WRITE (LOUT,101)
      WRITE (LOUT,465) T,TOUT,H
 465  FORMAT(//45H INDEX = 2 ON INPUT WITH (T - TOUT)*H .GE. 0./
     1        4H T =,E18.8,7H TOUT =,E18.8,4H H =,E18.8//)
      INDEX = -6
      RETURN
C
 500  T0 = T
      DO 510 I = 1,N
 510    Y0(I) = Y(I,1)
 520  INDEX = KFLAG
      TOP = T0
      H0 = HUSED
      IF (KFLAG .NE. 0) H0 = H
      RETURN
C----------------------- END OF SUBROUTINE DRIVE ---------------------
      END


      SUBROUTINE FMAX(A,N,B)
C
      DIMENSION A(1)
C
      C = A(1)
      DO 10 I = 2,N
      C = AMAX1(C,A(I))
   10 CONTINUE
      B = C
      RETURN
      END


      SUBROUTINE INTERP (TOUT, Y, NO, Y0)
C--------------------------------------------------------------------
C SUBROUTINE INTERP COMPUTES INTERPULATED VALUES OF THE DEPENDENT
C VARIABLE Y AND STORES THEM IN Y0.  THE INTERPOLATION IS TO THE
C POINT T = TOUT AND USES THE NORDSIECK HISTORY ARRAY Y AS FOLLOWS..
C                           NQ
C                 Y0(I)  =  SUM  Y(I,J+1)*S**J ,
C                           J=0
C WHERE S = -(T-TOUT)/H.
```

```
C-----------------------------------------------------------------------
      INTEGER NO
      INTEGER JSTART, KFLAG, MF, N
      INTEGER I, J, L
      DIMENSION YO(NO),Y(NO, 6)
C
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART
C(1
      DATA ONE /1.000/
      DO 10 I = 1,N
 10     YO(I) = Y(I,1)
      L = JSTART + 1
      S = (TOUT - T)/H
      S1 = ONE
      DO 30 J = 2,L
        S1 = S1*S
        DO 20 I = 1,N
 20       YO(I) = YO(I) + S1*Y(I,J)
 30     CONTINUE
      RETURN
C------------------------ END OF SUBROUTINE INTERP ---------------------
      END


      SUBROUTINE ISTATE
C
C     ----    SUBROUTINE ISTATE CALCULATES THE INITIAL CONCENTRATIONS
C             OF THE FOLLOWING SPECIES USING THE QUASI-STEADT STATE
C             APPROXIMATION.   THIS VERSION OF ISTATE IS TO BE USED
C             ONLY FOR THE ERT 30 SPECIES X 51 REACTION MODEL OF 4.1.78.
C
C             HONO    NO2     O       NO3     N2O5    AROH
C             ARCO    AO2     PA32    ARO     AO      PAN
C             RCO3    HO2     PAO     HNO4    O
C
      COMMON/CHEM1/ NOSTAT,             NOSTM1,             NOREAC,
     1              NOSPEC,             NSTDY,              NK
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),         RATKON(55),
     1              RATEFF(55),         RATEV(2,5),         QRATE,
     2              NVRATE,             LOCVRT(2)
      DIMENSION  C(40),  R(55),  A(2,2),  B(2)
      EQUIVALENCE (RATEFF, R)
      DATA YES /3HYES/
C
      DO 200 K=1,NOSTAT
C
      CALL XMIT(NOSPEC,CONIN(1,K),C)
      IF(QRATE.EQ.YES)  CALL RATEHI(K)
      CALL UNMIXR(K)
C
C     ------   NO2    ------
      C(2) = C(1)*C(3)*R(1)/R(3)
C     ------   O      ------
      C(25) = R(1)*C(2)/( R(2) + R(24)*C(8) )
```

```
C     ------   HONO   ---------
      AA = 2.*R(5)
      BB = R(6)
      CC = - 2.*R(4)*C(1)*C(2)*C(30) - R(7)*C(1)*C(24)
      C(4) = (-BB +(BB*BB -4.*AA*CC)**.5)/2./AA
C     ------   D    ------
      C(19) = (.5*R(22)*C(3)*C(8))/(R(23)*C(2) + R(25)*C(1))
C     ------  NO3 + N2O5   ------
      A(1,1) = R(14)*C(1) + R(15) *C(2)
      A(1,2) = -R(17)
      A(2,1) = -R(15)*C(2)
      A(2,2) = R(16)*C(30) + R(17)
      B(1) = R(13)*C(2)*C(3) + R(23)*C(2)*C(19)
      B(2) = 0.0
      CALL SOL2B2 (A,B)
      C(10) = B(1)
      C(27) = B(2)
C     ------     AROH    ------
      C(22) = R(41)*C(13)/R(42)
C     ------     ARCO    ------
      C(14) = R(42)*C(22)*C(24)/(R(43)*C(1))
C     ------     AO2     ------
      C(17) = R(18)*C(8)*C(24)/(R(19)*C(1))
C     ------      PAO2
      C(18) = R(26)*C(9)*C(24)/((R(27) + R(28))*C(1))
C     ------   HNO4 ------
      C(20) = R(11)*C(2)*C(11)/R(21)
C     ------     ARO     ------
      C(21) = R(44)*C(13)*C(24)/(R(45)*C(1))
C     ------     AO     ------
      C(23) = R(19)*C(1)*C(17)/R(20)
C     ------     PAN + RCO3    -----
      A(1,1) = R(37)
      A(1,2) = -R(35)*C(2)
      B(1) = 0.0
      A(2,1) = - R(37)
      A(2,2) = R(35)*C(1) + R(36)*C(2)
      B(2) = .25*R(22)*C(3)*C(8) + R(33)*C(7)*C(24)
      CALL SOL2B2(A,B)
      C(12) = B(1)
      C(16) = B(2)
C     ------   RO2 + PAO    ------
      A(1,1) = R(30)*C(1) + R(40)*C(11)
      A(1,2) = - R(29)
      B(1) = .4*R(24)*C(8)*C(25) + .15*R(27)*C(1)*C(18) + R(34)*C(7)
     1                                          + R(35)*C(1)*C(16)
      A(2,1) = -R(30)*C(1)
      A(2,2) = R(29) + R(31) + R(32)*C(2)
      B(2) = .85*R(27)*C(1)*C(18)
      CALL SOL2B2 (A,B)
      C(15) = B(1)
      C(26) = B(2)
      CALL XMIT (NOSPEC, C, CONIN(1,K)  )
200   CONTINUE
      RETURN
      END


      FUNCTION ITHOUR(TMIN)
C
C     CONVERTS TIME IN MINUTES FROM MIDNIGHT TO HOURS ON 24-HOUR CLOCK
C
      IA = IFIX(TMIN/.600 + 1.E-04)
      IB = IA/100
      IM = IFIX(TMIN+1.E-04) - 60*IB
      IF(IM.LT.60) GO  TO 10
      IM = IM - 60
      IB = IB + 1
   10 ITHOUR = IB*100 + IM
      RETURN
      END


      SUBROUTINE JACOB(A,C,R,N)
C
C     JACOB COMPUTES THE JACOBIAN OF THE CHEMICAL RATE EQUATIONS.
C
C     E?T PHOTOCHEMICAL MECHANISM (4.1.78)  (30 SPECIES X 51 REACTIONS)
C
      DIMENSION    A(N,N),       C(1),       R(1)
C
C     ZERO THE A ARRAY
      NN2 = N*N
      CALL XMIT(-NN2,0.0,A)
C
      A(  1,  1) = - R(  3)*C(  3) - R(  4)*C(  2)*C( 30) - R(  7)
     *    *C( 24) - R( 10)*C( 11) - R( 14)*C( 10) - R( 19)*C( 17) -
     *    R( 25)*C( 19) - R( 27)*C( 18) - R( 28)*C( 18) - R( 30)*C( 15)
     *     - R( 35)*C( 16) - R( 43)*C( 14) - R( 45)*C( 21)
      A(  1,  2) = + R(  1) - R(  4)*C(  1)*C( 30)
      A(  1,  3) = - R(  3)*C(  1)
      A(  1,  4) = +    2.00*R(  5)*C(  4) + R(  6)
      A(  1, 10) = - R( 14)*C(  1)
      A(  1, 11) = - R( 10)*C(  1)
      A(  1, 14) = - R( 43)*C(  1)
      A(  1, 15) = - R( 30)*C(  1)
      A(  1, 16) = - R( 35)*C(  1)
      A(  1, 17) = - R( 19)*C(  1)
      A(  1, 18) = - R( 27)*C(  1) - R( 28)*C(  1)
      A(  1, 19) = - R( 25)*C(  1)
      A(  1, 21) = - R( 45)*C(  1)
      A(  1, 24) = - R(  7)*C(  1)
C     A(  1, 30) = - R(  4)*C(  1)*C(  2)
      A(  2,  1) = + R(  3)*C(  3) - R(  4)*C(  2)*C( 30) + R( 10)
     *    *C( 11) +    2.00*R( 14)*C( 10) + R( 19)*C( 17) + R( 25)
     *    *C( 19) + R( 27)*C( 18) + R( 30)*C( 15) + R( 43)*C( 14) +
     *    R( 45)*C( 21)
      A(  2,  2) = - R(  1) - R(  4)*C(  1)*C( 30) - R(  8)*C( 24) -
     *    R( 11)*C( 11) - R( 13)*C(  3) - R( 15)*C( 10) - R( 23)*C( 19) KM  1324
     *     - R( 32)*C( 26) - R( 36)*C( 16)                              KM  1325
      A(  2,  3) = + R(  3)*C(  1) - R( 13)*C(  2)                      KM  1326
      A(  2,  4) = +     2.00*R(  5)*C(  4)                             KM  1327
      A(  2, 10) = +     2.00*R( 14)*C(  1) - R( 15)*C(  2)             KM  1328
      A(  2, 11) = + R( 10)*C(  1) - R( 11)*C(  2)                      KM  1329
      A(  2, 12) = + R( 37)                                             KM  1330
      A(  2, 14) = + R( 43)*C(  1)                                      KM  1331
      A(  2, 15) = + R( 30)*C(  1)                                      KM  1332
      A(  2, 16) = - R( 36)*C(  2)                                      KM  1333
      A(  2, 17) = + R( 19)*C(  1)                                      KM  1334
      A(  2, 18) = + R( 27)*C(  1)                                      KM  1335
      A(  2, 19) = - R( 23)*C(  2) + R( 25)*C(  1)                      KM  1336
      A(  2, 20) = + R( 21)                                             KM  1337
      A(  2, 21) = + R( 45)*C(  1)                                      KM  1338
      A(  2, 24) = - R(  8)*C(  2)                                      KM  1339
      A(  2, 26) = - R( 32)*C(  2)                                      KM  1340
      A(  2, 27) = + R( 17)                                             KM  1341
C     A(  2, 30) = - R(  4)*C(  1)*C(  2)                               KM  1342
      A(  3,  1) = - R(  3)*C(  3)                                      KM  1343
      A(  3,  2) = - R( 13)*C(  3)                                      KM  1344
      A(  3,  3) = - R(  3)*C(  1) - R( 13)*C(  2) - R( 22)*C(  8)      KM  1345
      A(  3,  8) = - R( 22)*C(  3)                                      KM  1346
      A(  3, 25) = + R(  2)                                             KM  1347
      A(  4,  1) = +     2.00*R(  4)*C(  2)*C( 30) + R(  7)*C( 24)      KM  1348
      A(  4,  2) = +     2.00*R(  4)*C(  1)*C( 30) +     0.15*R( 32)    KM  1349
     *    *C( 26)                                                       KM  1350
      A(  4,  4) = -     4.00*R(  5)*C(  4) - R(  6)                    KM  1351
      A(  4, 24) = + R(  7)*C(  1)                                      KM  1352
      A(  4, 26) = +     0.15*R( 32)*C(  2)                             KM  1353
C     A(  4, 30) = +     2.00*R(  4)*C(  1)*C(  2)                      KM  1354
      A(  5,  5) = - R(  9)*C( 24)                                      KM  1355
      A(  5,  6) = + R( 38) + R( 39)*C( 24)                             KM  1356
      A(  5,  7) = + R( 34)                                             KM  1357
      A(  5, 24) = - R(  9)*C(  5) + R( 39)*C(  6)                      KM  1358
      A(  6,  1) = +     0.50*R( 25)*C( 19)                             KM  1359
      A(  6,  2) = +     0.50*R( 23)*C( 19)                             KM  1360
      A(  6,  3) = +     0.50*R( 22)*C(  8)                             KM  1361
      A(  6,  6) = - R( 38) - R( 39)*C( 24)                             KM  1362
      A(  6,  8) = +     0.50*R( 22)*C(  3)                             KM  1363
      A(  6, 19) = +     0.50*R( 23)*C(  2) +     0.50*R( 25)*C(  1)    KM  1364
      A(  6, 23) = + R( 20)                                             KM  1365
      A(  6, 24) = - R( 39)*C(  6)                                      KM  1366
      A(  6, 26) = +     0.50*R( 29)                                    KM  1367
      A(  7,  1) = +     0.50*R( 25)*C( 19) + R( 43)*C( 14)             KM  1368
      A(  7,  2) = +     0.50*R( 23)*C( 19) +     0.15*R( 32)*C( 26)    KM  1369
      A(  7,  3) = +     0.50*R( 22)*C(  8)                             KM  1370
      A(  7,  7) = - R( 33)*C( 24) - R( 34)                             KM  1371
      A(  7,  8) = +     0.50*R( 22)*C(  3) +     0.30*R( 24)*C( 25)    KM  1372
      A(  7, 14) = + R( 43)*C(  1)                                      KM  1373
      A(  7, 19) = +     0.50*R( 23)*C(  2) +     0.50*R( 25)*C(  1) +  KM  1374
     *    R( 49)*C( 28)                                                 KM  1375
      A(  7, 23) = + R( 20)                                             KM  1376
      A(  7, 24) = - R( 33)*C(  7)                                      KM  1377
      A(  7, 25) = +     0.30*R( 24)*C(  8)                             KM  1378
```

```
  A(  7, 26) = +    0.50*R( 29) +    0.50*R( 31) +    0.15*R( 32)   KM  1379
 *    *C(  2)                                                      KM  1380
  A(  7, 28) = + R( 49)*C( 19)                                     KM  1381
  A(  8,  3) = - R( 22)*C(  8)                                     KM  1382
  A(  8,  8) = - R( 18)*C( 24) - R( 22)*C(  3) - R( 24)*C( 25)     KM  1383
  A(  8, 24) = - R( 18)*C(  8)                                     KM  1384
  A(  8, 25) = - R( 24)*C(  8)                                     KM  1385
  A(  9,  9) = - R( 26)*C( 24)                                     KM  1386
  A(  9, 24) = - R( 26)*C(  9)                                     KM  1387
  A( 10,  1) = - R( 14)*C( 10)                                     KM  1388
  A( 10,  2) = + R( 13)*C(  3) - R( 15)*C( 10) + R( 23)*C( 19)     KM  1389
  A( 10,  3) = + R( 13)*C(  2)                                     KM  1390
  A( 10, 10) = - R( 14)*C(  1) - R( 15)*C(  2)                     KM  1391
  A( 10, 19) = + R( 23)*C(  2)                                     KM  1392
  A( 10, 27) = + R( 17)                                            KM  1393
  A( 11,  1) = - R( 10)*C( 11) + R( 45)*C( 21)                     KM  1394
  A( 11,  2) = - R( 11)*C( 11)                                     KM  1395
  A( 11,  3) = +    0.25*R( 22)*C(  8)                             KM  1396
  A( 11,  5) = + R(  9)*C( 24)                                     KM  1397
  A( 11,  6) = +    0.67*R( 38) + R( 39)*C( 24)                    KM  1398
  A( 11,  7) = + R( 34)                                            KM  1399
  A( 11,  8) = +    0.25*R( 22)*C(  3) +    0.40*R( 24)*C( 25)     KM  1400
  A( 11, 11) = - R( 10)*C(  1) - R( 11)*C(  2) -    4.00*R( 12)    KM  1401
 *    *C( 11) - R( 40)*C( 15) - R( 48)*C( 28)                      KM  1402
  A( 11, 13) = + R( 41)*C( 24)                                     KM  1403
  A( 11, 15) = - R( 40)*C( 11)                                     KM  1404
  A( 11, 20) = + R( 21)                                            KM  1405
  A( 11, 21) = + R( 45)*C(  1)                                     KM  1406
  A( 11, 23) = + R( 20)                                            KM  1407
  A( 11, 24) = + R(  9)*C(  5) + R( 39)*C(  6) + R( 41)*C( 13)     KM  1408
  A( 11, 25) = +    0.40*R( 24)*C(  8)                             KM  1409
  A( 11, 26) = + R( 31)                                            KM  1410
  A( 11, 28) = - R( 48)*C( 11)                                     KM  1411
  A( 12,  2) = + R( 36)*C( 16)                                     KM  1412
  A( 12, 12) = - R( 37)                                            KM  1413
  A( 12, 16) = + R( 36)*C(  2)                                     KM  1414
  A( 13, 13) = - R( 41)*C( 24) - R( 44)*C( 24)                     KM  1415
  A( 13, 24) = - R( 41)*C( 13) - R( 44)*C( 13)                     KM  1416
  A( 14,  1) = - R( 43)*C( 14)                                     KM  1417
  A( 14, 14) = - R( 43)*C(  1)                                     KM  1418
  A( 14, 22) = + R( 42)*C( 24)                                     KM  1419
  A( 14, 24) = + R( 42)*C( 22)                                     KM  1420
  A( 15,  1) = +    0.15*R( 27)*C( 18) - R( 30)*C( 15) + R( 35)    KM  1421
 *    *C( 16)                                                      KM  1422
  A( 15,  7) = + R( 34)                                            KM  1423
  A( 15,  8) = +    0.40*R( 24)*C( 25)                             KM  1424
  A( 15, 11) = - R( 40)*C( 15)                                     KM  1425
  A( 15, 15) = - R( 30)*C(  1) - R( 40)*C( 11) - R( 50)*C( 28)     KM  1426
  A( 15, 16) = + R( 35)*C(  1)                                     KM  1427
  A( 15, 18) = +    0.15*R( 27)*C(  1)                             KM  1428
  A( 15, 25) = +    0.40*R( 24)*C(  8)                             KM  1429
  A( 15, 26) = + R( 29)                                            KM  1430
  A( 15, 28) = - R( 50)*C( 15)                                     KM  1431
  A( 16,  1) = - R( 35)*C( 16)                                     KM  1432
  A( 16,  2) = - R( 36)*C( 16)                                     KM  1433
```

```
      A( 16,  3) = +     0.25*R( 22)*C(  8)                             KM  1434
      A( 16,  7) = + R( 33)*C( 24)                                      KM  1435
      A( 16,  8) = +     0.25*R( 22)*C(  3)                             KM  1436
      A( 16, 12) = + R( 37)                                             KM  1437
      A( 16, 16) = - R( 35)*C(  1) - R( 36)*C(  2)                      KM  1438
      A( 16, 24) = + R( 33)*C(  7)                                      KM  1439
      A( 17,  1) = - R( 19)*C( 17)                                      KM  1440
      A( 17,  8) = + R( 18)*C( 24)                                      KM  1441
      A( 17, 17) = - R( 19)*C(  1) - R( 51)*C( 28)                      KM  1442
      A( 17, 24) = + R( 18)*C(  8)                                      KM  1443
      A( 17, 28) = - R( 51)*C( 17)                                      KM  1444
      A( 18,  1) = - R( 27)*C( 18) - R( 28)*C( 18)                      KM  1445
      A( 18,  9) = + R( 26)*C( 24)                                      KM  1446
      A( 18, 18) = - R( 27)*C(  1) - R( 28)*C(  1)                      KM  1447
      A( 18, 24) = + R( 26)*C(  9)                                      KM  1448
      A( 19,  1) = - R( 25)*C( 19)                                      KM  1449
      A( 19,  2) = - R( 23)*C( 19)                                      KM  1450
      A( 19,  3) = +     0.50*R( 22)*C(  8)                             KM  1451
      A( 19,  8) = +     0.50*R( 22)*C(  3)                             KM  1452
      A( 19, 19) = - R( 23)*C(  2) - R( 25)*C(  1) - R( 49)*C( 28)      KM  1453
      A( 19, 28) = - R( 49)*C( 19)                                      KM  1454
      A( 20,  2) = + R( 11)*C( 11)                                      KM  1455
      A( 20, 11) = + R( 11)*C(  2)                                      KM  1456
      A( 20, 20) = - R( 21)                                             KM  1457
      A( 21,  1) = - R( 45)*C( 21)                                      KM  1458
      A( 21, 13) = + R( 44)*C( 24)                                      KM  1459
      A( 21, 21) = - R( 45)*C(  1)                                      KM  1460
      A( 21, 24) = + R( 44)*C( 13)                                      KM  1461
      A( 22, 13) = + R( 41)*C( 24)                                      KM  1462
      A( 22, 22) = - R( 42)*C( 24)                                      KM  1463
      A( 22, 24) = + R( 41)*C( 13) - R( 42)*C( 22)                      KM  1464
      A( 23,  1) = + R( 19)*C( 17)                                      KM  1465
      A( 23, 17) = + R( 19)*C(  1) + R( 51)*C( 28)                      KM  1466
      A( 23, 23) = - R( 20)                                             KM  1467
      A( 23, 28) = + R( 51)*C( 17)                                      KM  1468
      A( 24,  1) = - R(  7)*C( 24) + R( 10)*C( 11)                      KM  1469
      A( 24,  2) = - R(  8)*C( 24)                                      KM  1470
      A( 24,  3) = +     0.50*R( 22)*C(  8)                             KM  1471
      A( 24,  4) = + R(  6)                                             KM  1472
      A( 24,  5) = - R(  9)*C( 24)                                      KM  1473
      A( 24,  6) = - R( 39)*C( 24)                                      KM  1474
      A( 24,  7) = - R( 33)*C( 24)                                      KM  1475
      A( 24,  8) = - R( 18)*C( 24) +     0.50*R( 22)*C(  3)             KM  1476
      A( 24,  9) = - R( 26)*C( 24)                                      KM  1477
      A( 24, 11) = + R( 10)*C(  1) + R( 48)*C( 28)                      KM  1478
      A( 24, 13) = - R( 41)*C( 24) - R( 44)*C( 24)                      KM  1479
      A( 24, 22) = - R( 42)*C( 24)                                      KM  1480
      A( 24, 24) = - R(  7)*C(  1) - R(  8)*C(  2) - R(  9)*C(  5) -    KM  1481
     *   R( 18)*C(  8) - R( 26)*C(  9) - R( 33)*C(  7) - R( 39)*C(  6)  KM  1482
     *    - R( 41)*C( 13) - R( 42)*C( 22) - R( 44)*C( 13) - R( 47)      KM  1483
     *   *C( 28)                                                        KM  1484
      A( 24, 28) = - R( 47)*C( 24) + R( 48)*C( 11)                      KM  1485
      A( 25,  2) = + R(  1)                                             KM  1486
      A( 25,  8) = - R( 24)*C( 25)                                      KM  1487
      A( 25, 25) = - R(  2) - R( 24)*C(  8) - R( 46)*C( 28)             KM  1488
      A( 25, 28) = - R( 46)*C( 25)                                      KM  1489
      A( 26,  1) = +    0,85*R( 27)*C( 18) + R( 30)*C( 15)              KM  1490
      A( 26,  2) = - R( 32)*C( 26)                                      KM  1491
      A(26,15) = R(30)*C(1) + R(50)*C(28)                               KM  1492
      A( 26, 18) = +    0,85*R( 27)*C(  1)                              KM  1493
      A( 26, 26) = - R( 29) - R( 31) - R( 32)*C(  2)                    KM  1494
      A(26,28) = R(50)*C(15)                                            KM  1495
      A( 27,  2) = + R( 15)*C( 10)                                      KM  1496
      A( 27, 10) = + R( 15)*C(  2)                                      KM  1497
      A( 27, 27) = - R( 16)*C( 30) - R( 17)                             KM  1498
C     A( 27, 30) = - R( 16)*C( 27)                                      KM  1499
      A( 28, 11) = - R( 48)*C( 28)                                      KM  1500
      A( 28, 15) = - R( 50)*C( 28)                                      KM  1501
      A( 28, 17) = - R( 51)*C( 28)                                      KM  1502
      A( 28, 19) = - R( 49)*C( 28)                                      KM  1503
      A( 28, 24) = - R( 47)*C( 28)                                      KM  1504
      A( 28, 25) = - R( 46)*C( 28)                                      KM  1505
      A( 28, 28) = - R( 46)*C( 25) - R( 47)*C( 24) - R( 48)*C( 11) -    KM  1506
     *     R( 49)*C( 19) - R( 50)*C( 15) - R( 51)*C( 17)                KM  1507
      A( 29, 11) = + R( 48)*C( 28)                                      KM  1508
      A( 29, 15) = + R( 50)*C( 28)                                      KM  1509
      A( 29, 17) = + R( 51)*C( 28)                                      KM  1510
      A( 29, 19) = + R( 49)*C( 28)                                      KM  1511
      A( 29, 24) = + R( 47)*C( 28)                                      KM  1512
      A( 29, 25) = + R( 46)*C( 28)                                      KM  1513
      A( 29, 28) = + R( 46)*C( 25) + R( 47)*C( 24) + R( 48)*C( 11) +    KM  1514
     *     R( 49)*C( 19) + R( 50)*C( 15) + R( 51)*C( 17)                KM  1515
C     A( 30,  1) = - R(  4)*C(  2)*C( 30)                               KM  1516
C     A( 30,  2) = - R(  4)*C(  1)*C( 30)                               KM  1517
C     A( 30,  4) = +    2,00*R(  5)*C(  4)                              KM  1518
C     A( 30,  6) = + R( 39)*C( 24)                                      KM  1519
C     A( 30,  7) = + R( 33)*C( 24)                                      KM  1520
C     A( 30,  9) = + R( 26)*C( 24)                                      KM  1521
C     A( 30, 13) = + R( 44)*C( 24)                                      KM  1522
C     A( 30, 24) = + R( 26)*C(  9) + R( 33)*C(  7) + R( 39)*C(  6) +    KM  1523
C    *     R( 44)*C( 13)                                                KM  1524
C     A( 30, 27) = - R( 16)*C( 30)                                      KM  1525
C     A( 30, 30) = - R(  4)*C(  1)*C(  2) - R( 16)*C( 27)               KM  1526
      RETURN                                                            KM  1527
      END                                                               KM  1528


      SUBROUTINE KEMOD2(INTIM,TSTOP,NUMFLX,FLXUNT)                      KM  1529
C                                                                       KM  1530
C     -------  KEMOD2 IS THE SECONDARY DRIVER FOR THE CHEMICAL/         KM  1531
C              DIFFUSION MODULE OF THE ERT TRAJECTORY MODEL.            KM  1532
C                                                                       KM  1533
C              KEMOD2 CONTROLS ALL I/O EXCEPT INITIAL EMISSION INPUTS.  KM  1534
C              KEMOD2 CONTROLS THE UPDATING PROCESSES.                  KM  1535
C                                                                       KM  1536
C     -------  FEATURES OF KEMOD2                                       KM  1537
C                  INTEGRATION BY EPISODE PACKAGE                       KM  1538
C                  VARIABLE SIZE VERTICAL MESH                          KM  1539
C                  GENERALIZED BOUNDARY CONDITIONS                      KM  1540
```

```
C                  TIME + HEIGHT DEPENDENT PHOTODISSOCIATION RATES,
C                   WITH SKY CLEARNESS RATIOS
C                  TEMPERATURE DEPENDENT REACTION RATE CONSTANTS
C
      REAL INTIM
      INTEGER BCFLAG
      COMMON/CHEM1/ NOSTAT,             NOSTM1,             NOREAC,
     1              NOSPEC,             NSTDY,              NK
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),         RATKON(55),
     1              RATEFF(55),         RATEV(2,5),         QRATE,
     2              NVRATE,             LOCVRT(2)
      COMMON/CHEM3/ ZEE(5),             DELZ(4),            HTCELL(6),
     1              TDELZ(2) ,          DFINIT(6),          SCALOW(4),
     2              DCOF(5),            FLXWAL(40),         FLXDGE(40),
     3              BCFLAG(40),         DPRATE(10),         DEPOWR(10),
     4              LOCDPF(10),         NOPFLX ,            SCALUP(4)
      COMMON/CHEM4/ RATK1(100,5),       RATK2(100,5),       RLAT,
     1              RLONG,              TMZONE,             SUNTIM,
     2              HIRATE,             JDATE,              NRATE
      COMMON /CHEM5/  REACT(20,55),SPEC(40),LOCFLX(10),
     1                NASFLX,FLXW1(30),FLXW2(30)
      COMMON/INPUTS/TITLE(20),          IDATE(10),          NCURV
      COMMON/FLUXES/FLXIN( 7,200),      FLXTIM(200),        NFLUX
      COMMON /PS1/  TPASS(200),         PS(7,5,75),         FRACT(3),
     1              NPTSR,              NPSFLX,             LOCPSF(7)
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPOINT
      COMMON/EPCOM6/PW(4500)
      COMMON/PRPLOT/KPS(5) ,            KSYM(5),            GRCON(5,100)
     1    ,      TOUT(100),   VALMAX(5)
      COMMON/EPCOM9/HUSED,              NQUSED,             NSTEP,
     1              NFE,                NJE
      DIMENSION     YO(150),            PPM (40,5),         TEMP(25),
     1              TMTEMP(25)  ,       HTINV(100,8),       WORK(200),
     2              CLOUDT(25),         CLOUDF(25)         ,VFR(5,200),
     3              PTSR(7,200)         .
C
      EQUIVALENCE (YES,IES),  (RNEG,NEG)
      EQUIVALENCE (PPM ,CONIN)
      EQUIVALENCE (VFR,PW),  (PTSR,PW(1001))
C
      DATA LUP, LIN, LOUT, YES, RNEG  /1, 3, 6, 3HYES, 2HNO/
      DATA KSYM /1HN, 1H2, 1HO, 1HS, 1HC  /
      DATA RMOLE /4HMOLE/
      DATA NROW,MAXR,MAXMSH,MAXNK,NRHT,KTMAX /40,55,5,30,100,100/
      DATA NF,IERROR /21,3/
C
C------------ READ INPUTS FOR COMPUTING POLLUTANT CONCENTRATIONS
C
      READ(LIN,30) DELT
      READ(LIN,30) BIGSTP
      READ(LIN,30) EPS
      READ(LIN,30) UPOINT
      READ(LIN,30) PRNTIN
      READ(LIN,29) TIMOUT
      READ(LIN,29) SUNGEN
      READ(LIN,29) HIRATE
      READ(LIN,29) QCLOUD
      READ(LIN,29) VTEMP
      READ(LIN,29) SPUNCH
      READ(LIN,31) NUMSTP
      READ(LIN,31) NOREAC
      READ(LIN,31) NOSPEC
      READ(LIN,31) NSTDY
      READ(LIN,31) NOSTAT
      READ(LIN,30) (ZEE(I), I=1,NOSTAT)
C
      NK = NOSPEC -NSTDY
      NPUNCH = NK
      SUMTIM = UPDINT
      NOCHEM = YES
      QRATE = YES
      F = YES
      TSTOP2 = TSTOP
      KOK = 100
      NASFLX = NUMFLX
C
C     ---- NOSPEC IS THE TOTAL NUMBER OF SPECIE    (MAX = 40)
C     ---- NSTDY IS THE NUMBER OF SPECIE HELD CONSTANT OR IN STEADY STAT
C     ---- NK IS THE NUMBER OF SPECIE INTEGRATED    (MAXNK = 30)
C     ---- NOSTAT IS THE NUMBER OF VERTICAL STATIONS (MIN=4,MAX=5)
C     -----LOCFLX IS LOCATION  INDEX FOR AREA SOURCE EMISSION FLUXES
C     -----NUMFLX IS NUMBER OF AREA SOURCE EMISSION FLUXES
C     -----LOCPSF IS LOCATION  INDEX OF POINT SOURCE EMISSION FLUXES
C     -----NPSFLX IS NUMBER OF POINT SOURCE EMMISION FLUXES
C     -----LOCDPF IS LOCATION  INDEX FOR DEPOSITING FLUXES
C     -----NDPFLX IS NUMBER OF DEPOSITING FLUXES
C
C     --------------BOUNDARY CONDITION CODE FOR WALL-------------------
C
C     BCFLAG = 0   DC/DZ = PHI/KZ   WHERE   PHI = 0.  (DEFAULT CASE)
C                                   OR      PHI = AREA SOURCE EMISSION F
C                                   AND     KZ  = DIFFUSION COEFFICIENT
C
C     BCFLAG = 1   DC/DZ = -DPRATE*(PPM**DEPOWR)/KZ  DEPOSITING SPECIE
C     ------WHERE
C     -----DPRATE(I) IS THE DEPOSITION VELOCITY OF THE I-TH  SPECIE
C     -----IN METERS/MINUTE/(PPM**(DEPOWR-1))
C     -----INPUT AS POSITIVE QUANTITY FOR UPTAKE AT WALL
C
C     BCFLAG = 2    C = INITIAL CONCENTRATION FOR ALL TIME
C
C     ----------------------------------------------------------------
C
C     ------  ZERO PPM ARRAY  AND CHECK NK AND NOSTAT INPUTS
      NRC = MAXMSH*NROW
      CALL XMIT (-NRC,0.,CONIN)
      IF(NOSTAT.GE.4)  GO TO 200
      WRITE(LOUT,77) NOSTAT
      GO TO 800
  200 CONTINUE
```

```
      IF(NK.LE.MAXNK)  GO TO 210
      WRITE(LOUT,76) NOSPEC ,NSTDY, NK
      GO TO 800
  210 CONTINUE
      IF(ROCHEM.NE.YES)  GO TO 219
      NOPFLX = 0
      DO 215  I=1,NOSPEC
      READ(LIN,51)  SPEC(I),WTMOLE(I),BCFLAG(I)
      IF(BCFLAG(I) .EQ. 1)  NOPFLX = NOPFLX + 1
  215 CONTINUE
      IF(NUMFLX.GT.0) READ(LIN,31) (LOCFLX(I),I=1,NUMFLX)
      IF(NPSFLX.GT.0) READ(LIN,31) (LOCPSF(I),I=1,NPSFLX)
      IF(NOPFLX.GT.0) READ(LIN,53) (LOCDPF(I),DPRATE(I),DEPOWR(I),
     1  I=1,NOPFLX)
  219 CONTINUE
      DO 220 I = 1,NOSPEC
      READ(LIN,41) (PPM(I,J),J=1,NOSTAT)
220   CONTINUE
      IF(ROCHEM.NE.YES)  GO TO 230
      DO 222 I=1,NOREAC
      READ(LIN,30) RATKON(I)
222   CONTINUE
      DO 224 J=1,NOREAC
      READ(LIN,40) (REACT(I,J),I=1,20)
224   CONTINUE
C
230   CONTINUE
C
      TIME = INTIM
C
C
C     --------- READ OR GENERATE THE PHOTODISSOCIATION RATES
C
C     -------- THE PHOTODISSOCIATION RATES ARE GENERATED IF SUNGEN=YES
C     -------- THE PHOTODISSOCIATION RATES VARY WITH HEIGHT IF HIRATE=YES
C     -------- LOCVRT IS THE SPECIES INDEX FOR THE RATES (INPUT)
C
      READ(LIN,30) RLAT
      READ(LIN,30) RLONG
      READ(LIN,30) TMZONE
      READ(LIN,31) JDATE
      READ(LIN,31)  NVRATE
      READ(LIN,31) (LOCVRT(I),I=1,NVRATE)
C
      IF(SUNGEN.EQ.YES)  GO TO 244
      DO 243 KK=1,NVRATE
      NHT=NOSTAT
      IF(HIRATE.EQ.RNEG)  NHT=1
      DO 240  I=1,101
      IF(KK.GT.1)  GO TO 236
      READ(LIN,54) (RATK1(I,IZ),IZ=1,NHT)
      IF(RATK1(I,1) .LT.  0.0)  GO TO 243
      GO TO 240
  236 IF(KK.GT.2)  GO TO 242
      READ(LIN,54) (RATK2(I,IZ),IZ=1,NHT)
      IF(RATK2(I,1) .LT.  0.0)  GO TO 243
  240 CONTINUE
  242 WRITE(LOUT,15)
      GO TO 800
  243 CONTINUE
      NRATE = I - 1
      GO TO 245
  244 CONTINUE
C
      JSTART = ITHOUR( INTIM )
      JSTOP  = ITHOUR( FLXTIM(NFLUX) )
      CALL PHOTOD (JSTART,JSTOP,NOSTAT,ZEE)
  245 CONTINUE
      IK1 = 1
C
      IF(QCLOUD.EQ.RNEG)  GO TO 249
C     ------- INPUT SKY COVER FACTORS AND TIMES
C     ------- LAST SKY COVER UPDATE TIME (CLOUDT) MUST BE A NEGATIVE
      DO 246  I=1,26
      READ(LIN,59)  A, B
      IF(A.LT.0.0)  GO TO 248
      IF(I.GT.25)  GO TO 247
      CLOUDT(I) = A
      CLOUDF(I) = B
      NCLOUD = I
  246 CONTINUE
      GO TO 248
  247 WRITE(LOUT,57)
      GO TO 800
  248 CONTINUE
      ICLOUD = 1
      CLOUDY = CLOUDF(1)
  249 IF(QCLOUD.EQ.RNEG)  CLOUDY= 1.0
C     ------        THIS CALL TO UPRAT2 INITIALIZES THE VARIABLE RATES
      CALL UPRAT2(TIME,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)
      CALL RATEMI(1)
260   CONTINUE
C
C     ------------- READ TEMPERATURES IF VTEMP = YES
      ITEMP = 1
      NTEMP = 1
      TEMP(1) = 298.2
      IF(VTEMP.NE.YES)  GO TO 265
      DO 261 I = 1,26
      READ(LIN,59)  A, B
      IF(I.GT.25)   GO TO 262
      IF(A.LT.0.)  GO TO 265
      TMTEMP(I) = A
      TEMP(I) = B + 273.2
      NTEMP = I
  261 CONTINUE
  262 CONTINUE
      WRITE(LOUT,79)
      GO TO 800
  265 CONTINUE
```

```
      CALL TEMPR (ITEMP,INTIM,TEMP)
C
C     -------- READ TABLE OF DIFFUSIVITY UPDATE TIMES, MIXING HEIGHTS,
C              AND DIFFUSIVITY COEFFICIENTS.
C              MIXING HEIGHT INFORMATION IS OPTIONAL.
      K = NOSTAT+3
      DO 270 I = 1,101
      READ(LIN,41) A,B,(HTINV(I,J),J=3,K)
      IF (A.LT.0.) GO TO 280
      HTINV(I,1) = A
      HTINV(I,2) = B
270   CONTINUE
      PRINT 45
      GO TO 800
280   NHT = I-1
      IHT = 1
      KHT = 1
290   CONTINUE
C
C     INITIALIZE PLOT PARAMETERS
C      FOH
      KPS(1) = 1
      KPS(2) = 2
      KPS(3) = 3
      KPS(4) = 28
      KPS(5) = 5
      NCURV = 5
      CALL XMIT(-NCURV,0,0,VALMAX)
C
C      BANNER PAGE
C
      WRITE(LOUT,99)
C
C     ----- REGULATE THE AREA SOURCE EMISSION FLUX SCHEDULES
      IF(NUMFLX.EQ.0)  GO TO 299
      DO 297 I=1,NUMFLX
      DO 293 J=1,NFLUX
      WORK(J) = FLXIN(I,J)
  293 CONTINUE
      CALL SKEDUL(WORK,FLXTIM,NFLUX,NUPDAT,UPDINT,INTIM,TSTOP,-1.,F,-1)
      IF(F.EQ. YES)  GO TO 800
C     -----   CONVERT FLUXES FROM MASS OR MOLE FRACTION TO PPM
      ID = LOCFLX(I)
      WT = 1.E6
      IF(FLXUNT.NE.RMULE)   WT = 28.97E6 /WTMOLE(ID)
      DO 295 J=1,NUPDAT
      FLXIN(I,J) = WORK(J)*WT
  295 CONTINUE
  297 CONTINUE
C     -----------  THE NEW SCHEDULE OF FLUXES IS STORED IN FLXIN
      NFLUX = NUPDAT
      IFLXTM = 1
  299 CONTINUE
C     ------- INITIALIZE FLUXES
      NPMX = 5*MAXNK
      CALL  XMIT(-NPMX,0,0,PSR1)
      CALL  XMIT(-NPMX,0,0,PSR2)
      CALL  XMIT(-MAXNK,0,0,FLXW1)
      CALL  XMIT(-MAXNK,0,0,FLXW2)
      CALL XMIT(-NROW,0.,FLXWAL)
      CALL XMIT(-NROW,0.,FLXDGE)
      CALL UPFLX1(TIME,IFLXTM)
C
C     *****   GENERATE VERTICAL MESH PARAMETERS
C
      NOSTM1 = NOSTAT - 1
      DELZ(1) = ZEE(2) - ZEE(1)
      HTCELL(1) = DELZ(1)
      DO 300 K = 2,NOSTM1
      DELZ(K) = ZEE(K+1) - ZEE(K)
      HTCELL(K) = .5*(DELZ(K) + DELZ(K-1))
  300 CONTINUE
      HTCELL(NOSTAT) = DELZ(NOSTM1)
C
C      ***** CONVERT POINT SOURCE EMISSIONS FROM ABSOLUTE MASS
C            OR MOLES TO RATES ON A REGULAR UPDATE SCHEDULE.
C
C      ***** POINT SUURCE MASS IS BLED IN OVER (UP TO) THREE
C            UPDATE INTERVALS. THE FRACTION IN EACH INTERVAL
C            IS SET BY THE FRACT ARRAY (SEE KEMOD DATA STATEMENT).
      IF(NPSFLX.LE.0)  GO TO 307
      CALL XMIT(-2625,0,0,PS)
      A = 1.E6/UPDINT
      IF(FLXUNT.EQ.RMOLE)  GO TO 303
      DO 302 K = 1,NPSFLX
      ID =LOCPSF(K)
      WORK(K) = 28.97/WTMOLE(ID)
  302 CONTINUE
  303 CONTINUE
      B = INTIM - 0.499*UPDINT
      K4 = 0
      DO 305 K = 1,NPTSR
      K1 = IFIX( (TPASS(K)-B)/UPDINT) + 1
      K2 = K1 + 1
      K3 = K1 + 2
      K4 = MAX0(K4,K3)
      DO 305 J = 1,NOSTAT
      VF = A*VFR(J,K)/HTCELL(J)
      IF(J.EQ.1  .OR.  J.EQ.NOSTAT)  VF = 2.*VF
      DO 305 I = 1,NPSFLX
      C = VF*PTSR(I,K)
      IF(FLXUNT.NE.RMOLE) C = C*WORK(I)
      PS(I,J,K1) = PS(I,J,K1) + FRACT(1)*C
      PS(I,J,K2) = PS(I,J,K2) + FRACT(2)*C
      PS(I,J,K3) = PS(I,J,K3) + FRACT(3)*C
  305 CONTINUE
      NPTSR = K4
      TPASS(1) = INTIM
      DO 306 K = 2,NPTSR
      TPASS(K) = TPASS(K-1) + UPDINT                                    KM
  306 CONTINUE                                                          KM
      N = MAXNK*NOSTAT                                                  KM
      CALL XMIT(-N,0.0,PSRATE)                                          KM
      CALL XMIT(-2400,0.0,PW)                                           KM
  307 IPS = 1                                                           KM
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)                                 KM
C                                                                       KM
C     ******* PRINT INPUTS                                              KM
C                                                                       KM
      CALL NEWPAG(TITLE,0,IDATE)                                        KM
      WRITE(LOUT,16) INTIM,TSTOP,DELT,BIGSTP,NOREAC,NOSPEC,NSTDY,NOSTAT KM
      IF(NUMFLX.EQ.0)  GO TO 311                                        KM
      DO 308 I= 1,NUMFLX                                                KM
      ID = LOCFLX(I)                                                    KM
      WORK(I) = SPEC(ID)                                                KM
  308 CONTINUE                                                          KM
      CALL NEWPAG(TITLE,0,IDATE)                                        KM
      WRITE(LOUT,68)  (WORK(I) , I=1,NUMFLX)                            KM
      DO 310  J = 1, NFLUX                                              KM
      IF(MOD(J,50).NE.0)  GO TO 309                                     KM
      CALL NEWPAG (TITLE,0,IDATE)                                       KM
      WRITE(LOUT,68)  (WORK(I), I=1,NUMFLX)                             KM
  309 WRITE(LOUT,69) FLXTIM(J), (FLXIN(I,J), I=1,NUMFLX)                KM
      FLXTIM(J+1)=FLXTIM(J) + UPDINT                                    KM
  310 CONTINUE                                                          KM
  311 CONTINUE                                                          KM
      IF(NPSFLX.EQ.0)  GO TO 317                                        KM
      DO 312  K=1,NPSFLX                                                KM
      ID = LOCPSF(K)                                                    KM
      WORK(K) = SPEC(ID)                                                KM
  312 CONTINUE                                                          KM
      CALL NEWPAG(TITLE,0,IDATE)                                        KM
      WRITE(LOUT,83)  (WORK(K),K=1,NPSFLX)                              KM
      DO 314 K = 1,NPTSR                                                KM
      IF(MOD(K,9).NE.0)  GO TO 313                                      KM
      CALL NEWPAG(TITLE,0,IDATE)                                        KM
      WRITE(LOUT,83)  (WORK(KK),KK=1,NPSFLX)                            KM
  313 WRITE(LOUT,84)  TPASS(K), (PS(I,1,K),I=1,NPSFLX)                  KM
      DO 314 J =2,NOSTAT                                                KM
      WRITE(LOUT,85)  (PS(I,J,K),I=1,NPSFLX)                            KM
  314 CONTINUE                                                          KM
  317 CONTINUE                                                          KM
C                                                                       KM
      CALL NEWPAG(TITLE,0,IDATE)                                        KM
      WRITE(LOUT,37)                                                    KM
      DO 320 I=1,NOREAC                                                 KM
  319 WRITE(LOUT,38) I, (REACT(J,I),J=1,20), RATKON(I)                  KM
  320 CONTINUE                                                          KM
      CALL NEWPAG(TITLE,0,IDATE)                                        KM
      WRITE(LOUT,28)                                                    KM
      DO 330 J = 1,NOSPEC                                               KM
  325 WRITE(LOUT,27)  J,  SPEC(J), WTMOLE(J), BCFLAG(J)                 KM
  330 CONTINUE                                                          KM
C                                                                       KM
```

```
C     ------  SCHEDULE AND INITIALIZE DIFFUSIVITIES
C
      IF(NHT.EQ.1)   GO TO 360
      K = NOSTAT + 3
      TSTOP = HTINV(NHT,1)
      CALL XMIT (NHT,HTINV(1,1),WORK(1))
      DO 350 J =2,K
      CALL SKEDUL(HTINV(1,J),WORK,NHT,NINT,UPDINT,INTIM,TSTOP,-1.,F,1)
      IF(F.EQ.YES)  GO TO 800
350   CONTINUE
      NHT = NINT
      HTINV(1,1) = INTIM
      DO 355 I =2,NHT
      HTINV(I,1) = HTINV(I-1,1) + UPDINT
355   CONTINUE
      TSTOP = TSTOP2
360   CONTINUE
      K = NOSTAT + 1
      DO 370 I =1,K
      J = I + 2
      DFINIT(I) = HTINV(1,J)
370   CONTINUE
      CALL DIFCOF(NOSTAT)
C      ----     INITIALIZE RATEFF
      CALL XMIT(NOREAC,RATKON,RATEFF)
C
410   IPROD = NK*NOSTAT
      TDELZ(1) = 2./HTCELL(1)
      TDELZ(2) = 2./HTCELL(NOSTAT)
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,10)
      HTCELL(1) = HTCELL(1)/2.
      HTCELL(NOSTAT) = HTCELL(NOSTAT)/2.
      I = 1
      WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)
      DO 420 I = 2,NOSTAT
      WRITE(LOUT,43) DFINIT(I), DELZ(I-1)
      IF(I.EQ.NOSTAT)  GO TO 415
      WRITE(LOUT,42)  I, ZEE(I), HTCELL(I)
      GO TO 420
415   WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)
420   CONTINUE
      HTCELL(1) = HTCELL(1)*2.
      HTCELL(NOSTAT) = HTCELL(NOSTAT)*2.
C
C     ----- INITIALIZE INTERNALLY COMPUTED CONCENTRATIONS
      CALL ISTATE
C     -----------  INITIALIZE YO VECTOR OF CONCENTRATIONS FOR DRIVE
      DO 425 I=1,NOSTAT
      K= NK*(I-1) + 1
      CALL XMIT(NK,CONIN(1,I),YO(K))
  425 CONTINUE
      CALL NEWPAG(TITLE,0,IDATE)
      ITIM = ITHOUR(TIME)
      WRITE(LOUT,97)   INTIM, ITIM
      WRITE(LOUT,18)  (SPEC(J),J=1,NOSPEC)
      DO 450 K=1,NOSTAT
      WRITE(LOUT,32)  ZEE(K), ( PPM(J,K),J=1,NOSPEC)
450   CONTINUE
C
      IF(SPUNCH,NE,YES)  GO TO 454
C     ------  WRITE I.C. ON TAPE LUP FOR PUNCH OR FILE STORAGE
      WRITE(LUP,80) TITLE
      WRITE(LUP,81) TIME,  (ZEE(K),K=1,NOSTAT)
      NOELV = 1
      DO 453 K=1,NOELV
      WRITE(LUP,81) ( PPM(J,K),J=1,NPUNCH)
453   CONTINUE
454   CONTINUE
      IF(NCURV.EQ.0) GO TO 459
      KT = 1
      DO 455 I = 1,NCURV
      J = KPS(I)
      IF(J.NE.5)  GO TO 456
C     SCALE CO FOR PLOTTING
      GRCON(I,KT) = .10*PPM(J,1)
      IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)
      GO TO 455
456   CONTINUE
      GRCON(I,KT) = PPM(J,1)
      IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)
  455 CONTINUE
      TOUT(KT) = INTIM
  459 CONTINUE
C     ----------   INITIALIZE ADDITIONAL PARAMETERS FOR INTEGRATION
      NSTEP = 0
      N = IPROD
      KNTER = 0
      TPRINT = INTIM + PRNTIN -.01
      RECALL = YES
      UPD4 = UPDINT/4.0
C
C     ------------- THE INTEGRATION TIME CYCLE BEGINS HERE -------------
C
      CALL TIMEX(0,A,B)
  460 CONTINUE
      OLDTIM = TIME
      IF(RECALL .NE. YES)  GO TO 470
      INDEX = 1
      TOUTEP = OLDTIM + UPDINT/10.
      IF(KNTER .GT.1)   DELT = .01
  470 IF(NSTEP .GT. NUMSTP)  GO TO 800
      TLAST = TIME
C
C                   INTEGRATE BY EPISODE
C
      CALL DRIVE(N,TIME,DELT,YO,TOUTEP,EPS,IERROR,MF,INDEX,BIGSTP,KOK)
C
C
      IF(KOK.GE.0)  GO TO 475
      ISTOP = IES
      WRITE(LOUT,60)
      GO TO 480
475   IF(INDEX .LT. 0)  GO TO 800
      INDEX = 2
      KNTER = KNTER + 1
      IF(RECALL.NE. YES)   GO TO 480
C     MAKE SHORT CALL TO EPISODE (DRIVE) IF RECALL EQUALS YES
C     THEN RESET TOUTEP TO ACTUAL TOUT ON NEXT CALL
      TOUTEP = OLDTIM + UPDINT
      RECALL = RNEG
      GO TO 470
480   TOUTEP = OLDTIM + 2.*UPDINT
C     ************* UNPACK YO FOR CHECKING AND OUTPUT
      DO 500 I=1,NOSTAT
      K = NK*(I-1) + 1
      CALL XMIT(NK,YO(K),CONIN(1,I)  )
500   CONTINUE



620   CONTINUE
      CALL PRODUK(UPDINT)
C
      IF(TIME.LT.TPRINT)  GO TO 694
      TPRINT = TPRINT + PRNTIN
 650  CONTINUE
C
C     ------- OUTPUT
C
      CALL NEWPAG(TITLE,0,IDATE)
      IF(NDPFLX. EQ. 0)  GO TO 682
      DO 681 J=1,NDPFLX
      I= LOCDPF(J)
      SINK =- DPRATE(J)*(PPM(I,1)**DEPOWR(J))
  681 WRITE(10,88)  SPEC(I),  SINK
  682 CONTINUE
      ITIM = ITHOUR(TIME)
      WRITE(LOUT,91)  TIME, ITIM, RATEV(1,1), DFINIT(1)
      IF(NUMFLX.LE.0)  GO TO 656
      DO 655 I = 1,NUMFLX
      K = LOCFLX(I)
      WORK(I) = SPEC(K)
      WORK(I+10) = FLXWAL(K)*TDELZ(1)
655   CONTINUE
656   IF(NPSFLX.LE.0)  GO TO 661
      DO 660 I = 1,NPSFLX
      K = LOCPSF(I)
      WORK(I+20) = SPEC(K)
      WORK(I+30) = PSRATE(K,1)
660   CONTINUE
661   CONTINUE
      NMX = MAXO(NUMFLX,NPSFLX)
      NMV = NMX + 1
      NMX = NMX + 2
      K = 2
      DO 662 I = NMV,NMX
      WORK(I) = SPEC(K)
      WORK(I+10) = FLXWAL(K)*TDELZ(1)
      WORK(I+20) = SPEC(K)
      WORK(I+30) = PSRATE(K,1)
      K = 7
662   CONTINUE
      WRITE(LOUT,92)
      WRITE(LOUT,93)  (WORK(I),WORK(I+10)
     1 ,              WORK(I+20), WORK(I+30),I=1,NMX)
      WRITE(LOUT,94)
      WRITE(LOUT,9) (SPEC(J),J=1,NOSPEC)
      DO 690 K = 1,NOSTAT
      WRITE(LOUT,32) ZEE(K) , (PPM(J,K),J=1,NOSPEC)
  690 CONTINUE
      IF (TIMOUT.NE.YES)  GO TO 691
      WRITE(LOUT,66)  INDEX,HUSED,NQUSED,NSTEP,NFE,NJE
      NTSTP = NSTEP
      CALL TIMEX(NTSTP,A,B)
  691 CONTINUE
      IF(SPUNCH.NE.YES)  GO TO 694
C     ------  WRITE OUTPUT ON TAPE LUP FOR PUNCH OR FILE STORAGE
      WRITE(LUP,81) TIME
      DO 693 K=1,NOELV
      WRITE(LUP,81) (PPM(J,K),J=1,NPUNCH)
  693 CONTINUE
  694 CONTINUE
C
      IF(NCURV.EQ.0) GO TO 700
      KT = KT + 1
      IF(KT.GT.KTMAX) GO TO 700
      DO 695 I = 1,NCURV
      J = KPS(I)
      IF(J.NE.5)  GO TO 696
C     SCALE CO FOR PLOTTING
      GRCON(I,KT) = .10*PPM(J,1)
      IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)
      GO TO 695
696   CONTINUE
      GRCON(I,KT) = PPM(J,1)
      IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)
  695 CONTINUE
      IOUT(KT) = TIME
  700 CONTINUE
      IF(TIME.GE.(TSTOP-.01)) ISTOP = IES
      IF(ISTOP.EQ. IES)  GO TO 800
C
C     ------------ UPDATE EMISSION FLUXES
      IFLXTM = IFLXTM+1
      IF (IFLXTM.GT.NFLUX) IFLXTM = NFLUX
      CALL UPFLX1(TIME,IFLXTM)
      IPS = IPS + 1
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)
```

```
720   CONTINUE
C
C     ------- UPDATE SKY COVER FACTOR
      IF (GRATE.EQ.RNEG) GO TO 730
      IF(GCLOUD.EQ.RNEG)  GO TO 725
      IF(ICLOUD.GE.NCLOUD)  GO TO 725
      BB= CLOUDT(ICLOUD+1)  - UPD4
      IF(TIME.LT. BB)  GO TO 725
      ICLOUD = ICLOUD + 1
      CLOUDY = CLOUDF(ICLOUD)
      WRITE(10,75)  TIME ,  CLOUDY
725   CONTINUE
C
C     -----------  UPDATE THE VARIABLE PHOTODISSOCIATION RATE CONSTANTS
C
      IK1 = IK1+1
      IF (IK1.GT.NRATE) IK1 = NRATE
      ROLDK1 = RATEV(1,1)
      CALL UPRAT2(TIME,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)
      IF(RATEV(1,1).LT..2*ROLDK1.OR.RATEV(1,1).GT.5.*ROLDK1)RECALL=YES
730   CONTINUE
C
C     ----          UPDATE DIFFUSION COEFFICIENTS
C
      IF(NHT.EQ.1)  GO TO 770
      AA= HTINV(IHT,1) - UPD4
      IF(TIME.LT.AA)  GO TO 770
      IF(KHT.GE.NHT)  GO TO 770
      IHT = IHT + 1
      KHT = KHT + 1
      WRITE (11,46) TIME,HTINV(IHT,1),HTINV(IHT,2)
      K = NOSTAT+1
      DO 750 I = 1,K
      DFINIT(I) = HTINV(IHT,I+2)
750   CONTINUE
      WRITE(11,96) (DFINIT(I),I=1,K)
      CALL DIFCOF(NOSTAT)
      IF (IHT.GT.NHT) IHT = NHT
770   CONTINUE
C
C      -----  UPDATE TEMPERATURE DEPENDENT REACTION RATES
      IF(VTEMP.NE.YES)  GO TO 780
      IF(ITEMP.GE.NTEMP)  GO TO 780
      IP1 = ITEMP + 1
      BB = TMTEMP(IP1) - UPD4
      IF(TIME.LT.BB)  GO TO 780
      ITEMP = IP1
      CALL TEMPR(ITEMP,TIME,TEMP)
780   CONTINUE
C
      GO TO 460
C
C     ------------  EXIT FROM INTEGRATION TIME CYCLE ----------------
C
800   CONTINUE
```

```
      IF(F.EQ.YES)  RETURN                                              KM  2201
      IF(INDEX.LT.0)  WRITE(LOUT,74) TIME, INDEX                        KM  2202
      IF(NCURV.EQ.0) GO TO 805                                          KM  2203
      IF(KT.GT.KTMAX) KT = KTMAX                                        KM  2204
      CALL COPLOT(KT)                                                   KM  2205
805   CONTINUE                                                          KM  2206
      IF(NSTEP .GT. NUMSTP)  WRITE(LOUT,26)                             KM  2207
      RETURN                                                            KM  2208
C                                                                       KM  2209
C     FORMATS                                                           KM  2210
C                                                                       KM  2211
8     FORMAT (1H ,5X,A4,8H FLUX = ,2X,G15.8)                            KM  2212
9     FORMAT (1H0,4X,9HHEIGHT-M.,3X,A4,7(11X,A4)/,6X,8(11X,A4),/,       KM  2213
     1  6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/    ) KM  2214
13    FORMAT (1H0,  /,  34X,48HCONCENTRATION PROFILES-PARTS PER MILLION KM  2215
     1        //45X,5HTIME=,E12.5,8H MINUTES//10X,20HGROUND DIFFUSIVITY KM  2216
     2=,G15.8,10X, 1HK,I1,3H = ,G12.4,10X,1HK,I2,3H = ,G12.4)           KM  2217
14    FORMAT (1H0,10X,30HINITIAL DIFFUSION COEFFICIENTS//10X,52HVALUES SKM  2218
     1HOWN ARE FOR POINTS HALFWAY BETWEEN STATIONS///10X,7HSTATION,10X,1KM  2219
     25HHEIGHT (METERS),10X,11HDIFFUSIVITY,16X,7HDELTA Z,11X,11HCELL HEIKM  2220
     3GHT  )                                                            KM  2221
15    FORMAT (1H1,10X,84HNUMBER OF NO2 RATE CONSTANTS EXCEEDED THE DIMENKM  2222
     1SIONS OF VECTOR -RATK1-, JOB ABORTED.)                            KM  2223
16    FORMAT (1H0,//,55H  INPUT DATA FOR VERTICAL DIFFUSION WITH PHOTOCHKM  2224
     1EMISTRY  ///8X,15HINITIAL TIME IS,F25.2,8H MINUTES//8X13HFINAL TIMKM  2225
     1E IS,F27.2,8H MINUTES//8X25HINITIAL TIME STEP SIZE IS,F15.6,8H MINKM  2226
     3UTES  //,8X25HMAXIMUM TIME STEP SIZE IS ,F15.2,8H MINUTES ,//,8X  KM  2227
     322HNUMBER OF REACTIONS IS ,I18,                                   KM  2228
     4    //8X,20HNUMBER OF SPECIES IS ,I20,//,8X, 36HNUMBER OF SPECIES KM  2229
     5IN STEADY STATE IS,I4,                                            KM  2230
     6                //8X33HNUMBER OF VERTICAL MESH POINTS IS,I7,4X,33HKM  2231
     7INCLUDING THE GROUND AND THE EDGE )                               KM  2232
17    FORMAT (1H0,//13X,8(8X,A4),6X,13HRATE CONSTANT,/,13X,8(8X,A4),    KM  2233
     1 /,13X,8(8X,A4),/,13X,8(8X,A4),/,13X,8(8X,A4) )                   KM  2234
18    FORMAT(1H0,                                                       KM  2235
     1/////4X9HHEIGHT-M.,5X,A4,7(11X,A4)/,7X,8(11X,A4),/,7X,8(11X,A4),  KM  2236
     2  /,7X,8(11X,A4),/,7X,8(11X,A4)  )                                KM  2237
19    FORMAT (1H0,5X8HREACTION,25X55H- - - R E A C T A N T   S T O I C HKM  2238
     1 I O M E T R Y - - -,24X,15H(1/(PPM--MIN))  )                     KM  2239
20    FORMAT(1H0,5X,I3,2X,8F12.2,10X,G12.5)                             KM  2240
21    FORMAT (1H ,10X,8F12.2,/, 3(11X,8F12.2,/)  )                      KM  2241
22    FORMAT (1H0,//6X8HREACTION,26X53H- - - P R O D U C T   S T O I C HKM  2242
     1 I O M E T R Y - - -)                                             KM  2243
26    FORMAT (1H1,25(1H /),10X,36HNUMBER OF INTEGRATION STEPS EXCEEDED,2KM  2244
     14H LIMIT.  JOB TERMINATED.)                                       KM  2245
27    FORMAT(1H ,I3,3H.  ,A4,14X,F7.1,16X,I2)                           KM  2246
28    FORMAT(1H0,/6X,7HSPECIES,7X,16HMOLECULAR WEIGHT,5X,9HBOUNDARY     KM  2247
     1  11HCONDITION     ,/)                                            KM  2248
29    FORMAT (40X,A4)                                                   KM  2249
30    FORMAT (40X,F10.0)                                                KM  2250
31    FORMAT (40X,I10)                                                  KM  2251
32    FORMAT (1H0,2X,F7.2,8E15.5,/, 4(10X,8E15.5,/) )                   KM  2252
33    FORMAT (1H1,25(/),25X,20A4//25X,8HDATE -- ,A10,12X,10HCP TIME = ,FKM  2253
     110.2,5H  SEC)                                                     KM  2254
37    FORMAT(1H0,    20X 17HR E A C T I O N S ,55X,30HRATE CONSTANTS (1/KM  2255
      1(PPM--MIN))  /)
 38   FORMAT(1H ,3X,I3,2H.  ,20A4,8X,E12.4)
 39   FORMAT (11X,A4,5X,11HAT STATION ,I3,5X,25HNEGATIVE OR ZERO AT TIME
     1 ,E10.3,5X,18HCONCENTRATION IS  ,E10.3,5X,8HDELT IS ,E10.3)
 40   FORMAT (20A4)
 41   FORMAT (20X,5E10.4)
 42   FORMAT(1H0,10X,I5,15X,F10.2,55X,F10.2)
 43   FORMAT(1H0,50X,G15.8,10X,F10.2)
 44   FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8,30X,F10.2)
 45   FORMAT (1H1,10X,57HTOO MANY INVERSION HEIGHT ENTRIES IN TABLE.  JO
     1B ABORTED.)
 46   FORMAT (1H0,10X,50HINVERSION HEIGHT AND DIFFUSIVITY UPDATED AT TIM
     1E ,F10.2,5X,14HUPDATE TIME = ,F10.2,5X,9HINV HT = ,F10.2)
 47   FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8)
 51   FORMAT (40X,A4,6X,F10.2,I10)
 52   FORMAT(20X,5(I2,8X)/20X,5(I2,8X))
 53   FORMAT (40X,I10,2F10.0)
 54   FORMAT (20X,5E12.4)
 55   FORMAT (20X,10F6.2)
 57   FORMAT (1H1,10X,45HTOO MANY CLOUD COVER ENTRIES  --  JOB ABORTED)
 58   FORMAT(10(/),10X,73HDEPLETION OF  NO  HAS CAUSED FAILURE OF CHEMIC
     1AL MODEL.  CASE TERMINATED. )
 59   FORMAT (40X,2F10.0)
 60   FORMAT(1H1,/////,20X,41HJOB TERMINATED -- NEGATIVE CONCENTRATIONS
     1   30H OR DEPLETION OF NITRIC OXIDE         )
 66   FORMAT(1H09X20HINTEGRATION INDEX = ,I4,10X,18HLAST STEP SIZE =
     1 E10.3,20X,17HLAST ORDER USED = ,I3,//,10X,18HNUMBER OF STEPS = ,
     2 I6,10X,33HNUMBER OF FUNCTION EVALUATIONS = ,I4,10X,34H NUMBER OF
     3JACOBIAN EVALUATIONS = ,I4)
 68   FORMAT(1H0, 5X,70HREVISED AREA SOURCE FLUX SCHEDULE       ---
     1    PPM-METERS/MINUTE   ,//,6X,4HTIME,7(11X,A4),/,10X,,3(11X,A4))
 69   FORMAT(1H ,F10.1,1X,7E15.4)
 74   FORMAT (1H0,//,8H  TIME =, F8.2,5X,9HINDEX =    ,I2,
     1  36H  INTEGRATION TERMINATED BY EPISODE   )
 75   FORMAT (1H0,10X,33HCLEARNESS RATIO UPDATED AT TIME =   ,F8.2,5X,
     1    20HSKY COVER FACTOR  =    , F8.4)
 76   FORMAT(1H1,9X,41HJOB ABORTED BECAUSE OF TOO MANY SPECIES  , //
     1/,9H NOSPEC =  ,I3,5X,9H  NSTDY=   ,I3,5X,5H NK=   ,I3)
 77   FORMAT(1H1,58HJOB ABORTED BECAUSE OF TOO FEW VERTICAL STATIONS  ,/
     1//,10X,8HNGOSTAT = , I2)
 79   FORMAT (47H0  TOO MANY TEMPERATURES INPUT -- JOB ABORTED       )
 80   FORMAT (20A4)
 81   FORMAT ( 7(6E12.4,/))
 83   FORMAT (1H0,5X,50HPOINT SOURCE EMISSION RATES BY VERTICAL CELL
     1    10X,10H(PPM/MIN)  ,//,6X,4HTIME,2X,8(8X,A4))
 84   FORMAT (1H0,F9.1,4X,8E12.3)
 85   FORMAT (1H ,13X,8E12.3)
 88   FORMAT(1H ,5X,A4,8H SINK = ,2X,G15.8)
 91   FORMAT(1H0,/,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,
     1 17H ON 2400-CLOCK)   //,40X,24HSURFACE CELL PARAMETERS
     2 40X,30HNO2 PHOTODISSOCIATION RATE  =   ,F8.3,10H   (/MIN)  ,/,
     3 40X,30HDIFFUSIVITY COEFFICIENT      =   ,F8.1,13H   (M**2/MIN))
 92   FORMAT(1H0,39X,12HAREA SOURCES,2X,9H(PPM/MIN)3X,13HPOINT SOURCES
     1  2X,9H(PPM/MIN) )
 93   FORMAT(  10(/,43X,A4,3H = ,E11.3,8X,A4,3H = ,E11.3))
```

```
94    FORMAT(1H ,//,40X,43HCONCENTRATION PROFILES IN PARTS PER MILLION )
96    FORMAT(1H ,10X,6F15.2)
97    FORMAT(1H0,/,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,
     1 17H ON 2400-CLOCK)   //,
     1/,40X,51HINITIAL CONCENTRATION PROFILES IN PARTS PER MILLION/)
99    FORMAT(11(//),57X,24HTHE ERT TRAJECTORY MODEL ,/57X6(4H====),
     1  ,//,59X,20H(CODE DATE   4.1.78) ////63X12HDEVELOPED BY  ,//
     1  48X,41HENVIRONMENTAL RESEARCH & TECHNOLOGY, INC.  //
     1  58X,22HSANTA BARBARA DIVISION   )
      END



      SUBROUTINE MATMUL(A,B,R,N,M,L)
      DIMENSION A(1),B(1),R(1)
C
      IR = 0
      IK = -M
      DO 10 K=1,L
        IK = IK + M
        DO 10 J=1,N
          IR = IR + 1
          JI = J-N
          IB = IK
          R(IR) = 0.
          DO 10 I = 1,M
            JI = JI+N
            IB = IB + 1
            R(IR) = R(IR) + A(JI)*B(IB)
   10 CONTINUE
      RETURN
      END



      SUBROUTINE PEDERV(NO,Y,Q,NKM)
C
C          THIS SUBROUTINE CALCULATES THE BLOCK DIAGONAL JACOBIAN
C          MATRACIES BY CALLING JACOB FOR EACH STATION
C
      INTEGER BCFLAG
      COMMON/CHEM1/ NOSTAT,                NOSTM1,                NOREAC,
     1              NOSPEC,                NSTDY,                 NK
      COMMON/CHEM2/ CONIN(40,5),           WTMOLE(40),            RATKON(55),
     1              RATEFF(55),            RATEV(2,5),            QRATE,
     2              NVRATE,                LOCVRT(2)
      COMMON/CHEM3/ ZEE(5),                DELZ(4),               HTCELL(6),
     1              TDELZ(2) ,             DFINIT(6),             SCALOW(4),
     2              DCOF(5),               FLXWAL(40),            FLXDGE(40),
     3              BCFLAG(40),            DPRATE(10),            DEPOWR(10),
     4              LOCDPF(10),            NDPFLX ,               SCALUP(4)
      DIMENSION     Y(NO,1),               Q(NKM,NKM,1)
      DATA YES /3HYES/
      DO 20 I = 1,NOSTAT
      KS = NK*(I-1) + 1
      IF(GRATE .EQ. YES)  CALL RATEHI(I)
      CALL XMIT(NK,Y(KS,1), CONIN(1,I))
      CALL UNMIXR(I)
      CALL JACOB(Q(1,1,I) , CONIN(1,I), RATEFF, NK )
      DO 10 J=1,NK
      Q(J,J,I) = Q(J,J,I) - DCOF(I)
   10 CONTINUE
      IF(I .NE. 1)  GO TO 20
      IF(NDPFLX.LT.1)  GO TO 20
      DO 15 K=1,NDPFLX
      ID = LOCDPF(K)
      IF(DEPOWR(K).LT. .99)  GO TO 12
      IF(DEPOWR(K).GT.1.01)  GO TO 12
      Q(ID,ID,1) = Q(ID,ID,1) -TDELZ(1)*DPRATE(K)
      GO TO 15
   12 Q(ID,ID,1) = Q(ID,ID,1) -TDELZ(1)*DPRATE(K)*DEPOWR(K)*
     1                                    (Y(ID)**(DEPOWR(K)-1.))
   15 CONTINUE
   20 CONTINUE
      RETURN
      END



      SUBROUTINE PHOTOD(JSTART,JSTOP,NOSTAT,ZEE)
C
C        PHOTOD GENERATES THE PHOTODISSOCIATION RATES OF NO2 AND HCHO
C        AS A FUNCTION OF TIME AND HEIGHT.
C
      DIMENSION ZINP(10),  HTINP(11),  ZEE(1),  HTD(6),  NAM(2),
     1          RA(11,10), RB(11,10)
      COMMON /INPUTS/ TITLE(20),  IDATE(10),   NCURV
      COMMON /CHEM4/  RATK1(100,5),       RATK2(100,5),   RLAT,
     1                RLONG,              TMZONE,         SUNTIM,
     2                HIRATE,             JDATE,          NRATE
      EQUIVALENCE (KYES,YES)
C
      DATA ZINP /0.,10.,20.,30.,40.,50.,60.,70.,78.,86./
      DATA HTINP /0.,150.,360.,640.,980.,1380.,1840.,2350.,2910.,
     1           3530.,4210./
C
C        RA CONTAINS SURFACE AND ELEVATED NO2 CLEAR-SKY NORMAL
C           AEROSOL PHOTODISSOCIATION RATES
C
      DATA RA /.579,.614,.645,.675,.703,.729,.752,.772,.790,.808,.824,
     1         .574,.609,.640,.671,.700,.725,.748,.768,.787,.805,.821,
     2         .560,.596,.628,.659,.688,.715,.737,.758,.777,.795,.812,
     3         .535,.572,.605,.637,.667,.694,.717,.738,.758,.776,.794,
     4         .496,.534,.568,.601,.631,.659,.683,.704,.725,.745,.764,
     5         .438,.477,.512,.545,.576,.604,.628,.650,.673,.694,.713,
     6         .352,.391,.426,.459,.490,.517,.541,.564,.583,.608,.631,
     7         .231,.264,.295,.325,.353,.378,.400,.421,.443,.466,.489,
     8         .114,.133,.153,.174,.194,.212,.229,.246,.264,.283,.303,
     9         .025,.027,.030,.034,.037,.040,.043,.046,.050,.055,.060/
C
```

```
C     RB CONTAINS THE SURFACE HCHO RATES AND THE PERCENTAGE CHANGE
C        ABOVE THE SURFACE RATES AS A FUNCTION OF ELEVATION.
C
      DATA RB /2.18,8.00,15.8,23.7,31.6,38.3,46.0,51.8,3*58.3,
     1         2.15,8.00,15.9,23.8,31.7,38.5,46.3,52.2,3*58.7,
     2         2.05,8.30,16.4,24.6,32.8,39.8,47.9,54.0,3*60.8,
     3         1.88,8.80,17.4,26.1,34.6,42.0,50.6,57.2,3*64.5,
     4         1.64,9.50,18.8,28.2,37.5,45.5,54.8,62.1,3*70.2,
     5         1.32,10.2,20.9,31.3,41.7,50.7,61.1,69.6,3*78.9,
     6         .934,12.0,23.8,35.9,47.9,58.5,70.8,81.1,3*92.5,
     7         .507,13.6,27.4,41.7,56.3,69.5,84.6,98.6,3*114.,
     8         .211,13.5,27.7,42.8,58.7,73.3,90.2,107.,3*126.,
     9         .040,11.6,22.7,34.1,45.6,55.5,66.9,77.6,3*89.3/
      DATA NHTI /11/
      DATA NZINP /10/
      DATA NAM /4HNO2 , 4HHCHO /
      DATA KYES /4HYES /
C
C     SPECIFY THE RA & RB MULTIPLIERS
C
      DATA RADJST, RBEXP /0.950,.00285/
      DATA NV /5/
C
C
      IINC = IFIX(SUNTIM)
      TIME = FLOAT(JSTART)
      NTMSTP = (JSTOP - JSTART)*6/10/IINC + 1
      IY = JDATE/10000
      IM = JDATE/100 - IY*100
      ID = JDATE - IY*10000 - IM*100
      IY = IY + 1900
      IRITE = NV
      NSTAT = NOSTAT
      CALL XMIT(NSTAT,ZEE,HTD)
      IF(HIRATE.NE.YES)  NSTAT = 1
      DO 5 K =1,10
      RB(1,K) = RB(1,K)*RBEXP
      RA(1,K) = RADJST*RA(1,K)
      DO 5 J =2,11
      RB(J,K) = RB(1,K)*(1.0 + 0.01*RB(J,K))
      RA(J,K) = RADJST*RA(J,K)
    5 CONTINUE
C
      DO 50 IS = 1,NTMSTP
      CALL SOLAR(RLAT,RLONG,TMZONE,IY,IM,ID,TIME,D,NV)
      ZEN = 90. - D
      IF(ZEN.LE.ZINP(1))  GO TO 30
      IF(ZEN.GE.ZINP(NZINP))  GO TO 30
C        FIND NEIGHBORING ZENITH ANGLE INDICIES
      DO 10 I = 1,NZINP
      IF(ZINP(I).LT.ZEN)   GO TO 10
      Z1 = ZINP(I-1)
      Z2 = ZINP(I)
      I1 = I - 1
      I2 = I
      GO TO 15
   10 CONTINUE
   15 CONTINUE
C
C         INTERPOLATE ON ZENITH ANGLE THEN ELEVATION
C
      J = 1
      DO 20 JJ = 1,NSTAT
      P = (ZEN - Z1)/(Z2 - Z1)
      K = J + 1
      R1 = RA(J,I1) + P*(RA(J,I2) - RA(J,I1))
      R2 = RA(K,I1) + P*(RA(K,I2) - RA(K,I1))
      PH = (HTD(JJ) - HTINP(J))/(HTINP(K) - HTINP(J))
      RATK1(IS,JJ) = R1 + PH*(R2 - R1)
      R1 = RB(J,I1) + P*(RB(J,I2) - RB(J,I1))
      R2 = RB(K,I1) + P*(RB(K,I2) - RB(K,I1))
      RATK2(IS,JJ) = R1 + PH*(R2 - R1)
      IF(JJ.EQ.NSTAT)   GO TO 20
      DO 18 KK = 1,5
      IF(HTD(JJ+1) .GT. HTINP(J+1))  J = J + 1
   18 CONTINUE
      IF(J.LE.NHTI)  GO TO 20
      J = NHTI
      HTD(JJ+1) = HTINP(NHTI)
   20 CONTINUE
C
      GO TO 40
   30 CONTINUE
      DO 35 JJ = 1,NSTAT
      RATK1(IS,JJ) = 5.E-03
      RATK2(IS,JJ) = 1.E-05
   35 CONTINUE
   40 CONTINUE
C
C         UPDATE TIME ON 2400 HR CLOCK
C
      IHR = TIME/100
      MIN = TIME - IHR*100
      NM = MIN + IINC
      IF(NM.LT.60)  GO TO 45
      NM = NM - 60
      IHR = IHR + 1
   45 TIME = IHR*100 + NM
   50 CONTINUE
C
C        WRITE PHOTODISSOCIATION RATE ARRAYS  (IF IRITE EQUALS YES)
C
      IF(IRITE.NE.KYES)  GO TO 80
      K = 1
   55 CONTINUE
      IHR = JSTART/100
      MIN = JSTART - IHR*100
      TIME = FLOAT( IHR*60 + MIN )
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(6,120)  NAM(K) ,  (HTD(I),I=1,NSTAT)
      DO 70 J = 1,NTMSTP                                                KM  2522
      IF(MOD(J,50) .NE. 0)  GO TO 60                                    KM  2523
      CALL NEWPAG(TITLE,0,IDATE)                                        KM  2524
      WRITE(6,120)  NAM(K) ,  (HTD(I),I=1,NSTAT)                        KM  2525
   60 CONTINUE                                                          KM  2526
      IF(K.EQ.1)  WRITE(6,130) TIME,(RATK1(J,I),I=1,NSTAT)              KM  2527
      IF(K.EQ.2)  WRITE(6,130) TIME,(RATK2(J,I),I=1,NSTAT)              KM  2528
      TIME = TIME + SUNTIM                                              KM  2529
   70 CONTINUE                                                          KM  2530
      K = K + 1                                                         KM  2531
      IF(K.LE.2)  GO TO 55                                              KM  2532
   80 NRATE = NTMSTP                                                    KM  2533
      RETURN                                                            KM  2534
C                                                                       KM  2535
  120 FORMAT (1H0,A4,1X,24HPHOTODISSOCIATION RATES ,20X,10HELEVATIONS,  KM  2536
     1         //,6X,4HTIME,10X,5(F10.1,2H M) ,/)                       KM  2537
  130 FORMAT (1H ,4X,F5.0,10X,5F12.5)                                   KM  2538
      END                                                               KM  2539


      SUBROUTINE PRODUK(UPDINT)                                         KM  2540
C                                                                       KM  2541
C         *PRODUK* CAN BE MODIFIED TO CALCULATE                         KM  2542
C          APPROXIMATE PRODUCT SPECIES CONCENTRATIONS                   KM  2543
C                                                                       KM  2544
      COMMON/CHEM1/ NOSTAT,NOSTM1,NOREAC,NOSPEC,NSTDY,NK                KM  2545
      COMMON/CHEM2/ CONIN(40,5),WTMOLE(40),RATKON(55),RATEFF(55),       KM  2546
     1  RATEV(2,5),QRATE,NVRATE,LOCVRT(2)                               KM  2547
C                                                                       KM  2548
C     ----- CURRENTLY *PRODUK* IS A DUMMY SUBROUTINE  ----              KM  2549
C                                                                       KM  2550
      RETURN                                                            KM  2551
      END                                                               KM  2552


      SUBROUTINE PSET (Y,NO,CON,MITER,IER)                              KM  2553
C-----------------------------------------------------------------------KM  2554
C     THIS VERSION OF PSET IS DESIGNED FOR A BLOCK TRI-DIAGONAL         KM  2555
C     SYSTEM OF ODE*S AND IS ONLY TO BE USED WITH MITER = 1.            KM  2556
C     MODIFICATIONS BY F.W. LURMANN       (7.8.77).                     KM  2557
C-----------------------------------------------------------------------KM  2558
C PSET IS CALLED BY TSTEP TO COMPUTE AND TO PROCESS THE MATRIX          KM  2559
C P = I - (H/EL(2))*J, WHERE J IS AN APPROXIMATION TO THE               KM  2560
C JACOBIAN.  J IS COMPUTED BY THE USER SUPPLIED SUBROUTINE PEDERV.      KM  2561
C-----------------------------------------------------------------------KM  2562
      INTEGER IER, MITER, NO                                            KM  2563
      INTEGER IPIV, JSTART, KFLAG, MF, N, NSQ                           KM  2564
      INTEGER I, IER, J, J1, N                                          KM  2565
      DIMENSION Y(NO,1)                                                 KM  2566
C                                                                       KM  2567
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART     KM  2568
      COMMON /EPCOM2/ YMAX(1)                                           KM  2569
      COMMON /EPCOM4/ SAVE1(1)                                          KM  2570
      COMMON /EPCOM5/ SAVE2(1)
      COMMON /EPCOM6/ PW(1)
      COMMON /EPCOM7/ IPIV(1)
      COMMON /EPCOM8/ EPSJ,NSQ
      COMMON /EPCM12/ PWL(3600),  WORKR(30), DUPPER(4), DLOWER(4)
      INTEGER BCFLAG
      COMMON/CHEM1/ NOSTAT,                NOSTM1,                NOREAC,
     1              NOSPEC,                NSTDY,                 NK
      COMMON/CHEM2/ CONIN(40,5),           WTMOLE(40),            RATKON(55),
     1              RATEFF(55),            RATEV(2,5),            QRATE,
     2              NVRATE,                LOCVRT(2)
      COMMON/CHEM3/ ZEE(5),                DELZ(4),               HTCELL(6),
     1              TDELZ(2) ,             DFINIT(6),             SCALOW(4),
     2              DCOF(5),               FLXWAL(40),            FLXDGE(40),
     3              BCFLAG(40),            DPRATE(10),            DEPOWR(10),
     4              LOCDPF(10),            NDPFLX ,               SCALUP(4)
      DATA ONE /1.0E0/, REP /1.0E-3/, ZERO /0.0E0/
      NKM = NK
      IF (MITER.NE.1)  GO TO 9
C IF MITER = 1, CALL PEDERV AND MULTIPLY BY A SCALAR.  -----------------
      CALL PEDERV(NO,Y,PW,NKM)
C     SCALE BLOCK DIAGONAL MATRICIES Q BY CON
      NK2N = NK*NK*NOSTAT
      DO 5 I=1,NK2N
      PW(I) = PW(I)*CON
    5 CONTINUE
C     ADD IDENTITY MATRIX TO Q MATRICES
      NKP1 = NK + 1
      NK2 = NK*NK
      DO 6 I=1,NOSTAT
      K = (I-1)*NK2 + 1
      DO 6 J=1,NK
      PW(K) = PW(K) + ONE
      K = K + NKP1
    6 CONTINUE
C     SCALE UPPER AND  LOWER OFF-DIAGONAL VECTORS BY CON
      DO 7 K=1,NOSTM1
      DUPPER(K) = SCALUP(K)*CON
      DLOWER(K) = SCALOW(K)*CON
    7 CONTINUE
      CALL BLKDEC(PW,NK,NOSTAT,IPIV,DLOWER,DUPPER,PWL,WORKR,NOGO,BCFLAG)
C     NOW TL AND Q CONTAIN THE LOWER AND UPPER COMPOSITION OF Q
C     AND IPIV CONTAINS THE PIVOTING INFORMATION FOR BLKSOL
      IER = NOGO
      RETURN
    9 CONTINUE
20    WRITE(6,21)
21    FORMAT (1H0,90H THIS VERSION OF PSET IS RESTRICTED TO MITER = 1
     1 ---    JOB ABORTED                                                )
      STOP
22    RETURN
C----------------------- END OF SUBROUTINE PSET ------ ----------------
      END
      SUBROUTINE RATEHI(K)                                              KM  2624
C                                                                       KM  2625
C     -------- THIS SUBROUTINE TRANSFERS THE VARIABLE RATE              KM  2626
C              CONSTANTS AT HIGHER ELEVATIONS  IN  THE ARRAY            KM  2627
C              RATEV TO RATKON AND RATEFF                               KM  2628
C     -------- PHOTODISSOCIATION RATES WHICH ARE PRPORTIONAL            KM  2629
C              TO  RATEV(1,N) OR RATEV(2,N)ARE ASSIGNED                 KM  2630
C     -------- HONO AND RCHO PHOTOLYSIS RATES ARE ASSUMED               KM  2631
C              PROPORTIONAL TO NO2 AND HCHO.                            KM  2632
C     -------- THIS ROUTINE IS SPECFIC TO THE ERT                       KM  2633
C              30 SPECIE X 51 REACTION MECHANISM 4.1.78                 KM  2634
C                                                                       KM  2635
      COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),        RATKON(55), KM  2636
     1              RATEFF(55),        RATEV(2,5),        QRATE,      KM  2637
     2              NVRATE,            LOCVRT(2)                      KM  2638
      I1 = LOCVRT(1)                                                    KM  2639
      RATKON(I1) = RATEV( 1,K)                                          KM  2640
      RATEFF(I1) = RATKON(I1)                                           KM  2641
      RATKON( 6) = .280*RATKON(I1)                                      KM  2642
      RATEFF( 6) = RATKON( 6)                                           KM  2643
      IF(NVRATE.LT. 2)  RETURN                                          KM  2644
      I2 = LOCVRT(2)                                                    KM  2645
      RATKON(I2)  = RATEV(2,K)                                          KM  2646
      RATEFF(I2)  =  RATKON(I2)                                         KM  2647
C     ALDEHYDE PHOTOLYSIS CORRECTION                                    KM  2648
      RATKON(34) = RATKON(I2)/3.0                                       KM  2649
      RATEFF(34) = RATKON(34)                                           KM  2650
      RETURN                                                            KM  2651
      END                                                               KM  2652


      SUBROUTINE RATES(Y,YDOT)                                          KM  2653
C                                                                       KM  2654
C          CALCULATION OF CHEMICAL RATES                                KM  2655
C                                                                       KM  2656
      INTEGER BCFLAG                                                    KM  2657
      COMMON/CHEM1/ NOSTAT,            NOSTM1,            NOREAC,     KM  2658
     1              NOSPEC,            NSTDY,             NK          KM  2659
      COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),        RATKON(55), KM  2660
     1              RATEFF(55),        RATEV(2,5),        QRATE,      KM  2661
     2              NVRATE,            LOCVRT(2)                      KM  2662
      COMMON/CHEM3/ ZEE(5),            DELZ(4),           HTCELL(6),  KM  2663
     1              TDELZ(2) ,         DFINIT(6),         SCALOW(4),  KM  2664
     2              DCOF(5),           FLXWAL(40),        FLXDGE(40), KM  2665
     3              BCFLAG(40),        DPRATE(10),        DEPOWR(10), KM  2666
     4              LOCDPF(10),        NDPFLX ,           SCALUP(4)   KM  2667
      DIMENSION Y(1), YDOT(1)  ,  RATE(40),  C(40),  R(55)              KM  2668
      EQUIVALENCE (R,RATEFF)                                            KM  2669
      DATA  YES  /3HYES/                                                KM  2670
      NKP1 = NK + 1                                                     KM  2671
      DO 130 K = 1,NOSTAT                                               KM  2672
      KS = NK*(K-1)                                                     KM  2673
      DO 90  I=1,NK                                                     KM  2674
      C(I) = Y(I+KS)                                                    KM  2675
```

```
   90 CONTINUE
      IF(NSTDY.EQ.0)  GO TO 105
      DO 100 I=NKP1,NOSPEC
      C(I) = CUNIN(I,K)
  100 CONTINUE
  105 CONTINUE
C
      IF(QRATE.EQ.YES)  CALL RATEHI(K)
      CALL UNMIXR(K)
C
C      EXPLICIT CHEMICAL RATE EQUATIONS FOR
C
C    ERT PHOTUCHEMICAL MECHANISM (4.1.78)  (30 SPECIES X 51 REACTIONS)
C
       RATE(  1) = + R(  1)*C(  2) - R(  3)*C(  1)*C(  3) - R(  4)*C(  1)
     *     *C(  2)*C( 30) + R(  5)*C(  4)** 2 + R(  6)*C(  4) - R(  7)
     *     *C(  1)*C( 24) - R( 10)*C(  1)*C( 11) - R( 14)*C(  1)*C( 10)
     *      - R( 19)*C(  1)*C( 17) - R( 25)*C(  1)*C( 19) - R( 27)*C(  1)
     *     *C( 18) - R( 28)*C(  1)*C( 18) - R( 30)*C(  1)*C( 15) - R( 35)
     *     *C(  1)*C( 16) - R( 43)*C(  1)*C( 14) - R( 45)*C(  1)*C( 21)
       RATE(  2) = - R(  1)*C(  2) + R(  3)*C(  1)*C(  3) - R(  4)*C(  1)
     *     *C(  2)*C( 30) + R(  5)*C(  4)** 2 - R(  8)*C(  2)*C( 24) +
     *     R( 10)*C(  1)*C( 11) - R( 11)*C(  2)*C( 11) - R( 13)*C(  2)
     *     *C(  3) +  2.00*R( 14)*C(  1)*C( 10) - R( 15)*C(  2)*C( 10) +
     *     R( 17)*C( 27) + R( 19)*C(  1)*C( 17) + R( 21)*C( 20) - R( 23)
     *     *C(  2)*C( 19) + R( 25)*C(  1)*C( 19) + R( 27)*C(  1)*C( 18)
     *      + R( 30)*C(  1)*C( 15) - R( 32)*C(  2)*C( 26) - R( 36)*C(  2)
     *     *C( 16) + R( 37)*C( 12) + R( 43)*C(  1)*C( 14) + R( 45)*C(  1)
     *     *C( 21)
       RATE(  3) = + R(  2)*C( 25) - R(  3)*C(  1)*C(  3) - R( 13)*C(  2)
     *     *C(  3) - R( 22)*C(  3)*C(  8)
       RATE(  4) = +  2.00*R(  4)*C(  1)*C(  2)*C( 30) -  2.00*R(  5)
     *     *C(  4)** 2 - R(  6)*C(  4) + R(  7)*C(  1)*C( 24) +
     *      0.15*R( 32)*C(  2)*C( 26)
       RATE(  5) = - R(  9)*C(  5)*C( 24) + R( 34)*C(  7) + R( 38)*C(  6)
     *      + R( 39)*C(  6)*C( 24)
       RATE(  6) = + R( 20)*C( 23) +  0.50*R( 22)*C(  3)*C(  8) +
     *      0.50*R( 23)*C(  2)*C( 19) +  0.50*R( 25)*C(  1)*C( 19) +
     *      0.50*R( 29)*C( 26) - R( 38)*C(  6) - R( 39)*C(  6)*C( 24)
       RATE(  7) = + R( 20)*C( 23) +  0.50*R( 22)*C(  3)*C(  8) +
     *      0.50*R( 23)*C(  2)*C( 19) +  0.30*R( 24)*C(  8)*C( 25) +
     *      0.50*R( 25)*C(  1)*C( 19) +  0.50*R( 29)*C( 26) +
     *      0.50*R( 31)*C( 26) +  0.15*R( 32)*C(  2)*C( 26) - R( 33)
     *     *C(  7)*C( 24) - R( 34)*C(  7) + R( 43)*C(  1)*C( 14) + R( 49)
     *     *C( 19)*C( 28)
       RATE(  8) = - R( 18)*C(  8)*C( 24) - R( 22)*C(  3)*C(  8) - R( 24)
     *     *C(  8)*C( 25)
       RATE(  9) = - R( 26)*C(  9)*C( 24)
       RATE( 10) = + R( 13)*C(  2)*C(  3) - R( 14)*C(  1)*C( 10) - R( 15)
     *     *C(  2)*C( 10) + R( 17)*C( 27) + R( 23)*C(  2)*C( 19)
       RATE( 11) = + R(  9)*C(  5)*C( 24) - R( 10)*C(  1)*C( 11) - R( 11)
     *     *C(  2)*C( 11) -  2.00*R( 12)*C( 11)** 2 + R( 20)*C( 23) +
     *     R( 21)*C( 20) +  0.25*R( 22)*C(  3)*C(  8) +  0.40*R( 24)
     *     *C(  8)*C( 25) + R( 31)*C( 26) + R( 34)*C(  7) +  0.67*R( 38)
     *     *C(  6) + R( 39)*C(  6)*C( 24) - R( 40)*C( 11)*C( 15) + R( 41)
     *    *C( 13)*C( 24) + R( 45)*C(  1)*C( 21) - R( 48)*C( 11)*C( 28)  KM  2731
      RATE( 12) = + R( 36)*C(  2)*C( 16) - R( 37)*C( 12)                KM  2732
      RATE( 13) = - R( 41)*C( 13)*C( 24) - R( 44)*C( 13)*C( 24)         KM  2733
      RATE( 14) = + R( 42)*C( 22)*C( 24) - R( 43)*C(  1)*C( 14)         KM  2734
      RATE( 15) = +  0.40*R( 24)*C(  8)*C( 25) +  0.15*R( 27)*C(  1)    KM  2735
     *    *C( 18) + R( 29)*C( 26) - R( 30)*C(  1)*C( 15) + R( 34)*C(  7)KM  2736
     *     + R( 35)*C(  1)*C( 16) - R( 40)*C( 11)*C( 15) - R( 50)*C( 15)KM  2737
     *    *C( 28)                                                       KM  2738
      RATE( 16) = +  0.25*R( 22)*C(  3)*C(  8) + R( 33)*C(  7)*C( 24) - KM  2739
     *    R( 35)*C(  1)*C( 16) - R( 36)*C(  2)*C( 16) + R( 37)*C( 12)   KM  2740
      RATE( 17) = + R( 18)*C(  8)*C( 24) - R( 19)*C(  1)*C( 17) - R( 51)KM  2741
     *    *C( 17)*C( 28)                                                KM  2742
      RATE( 18) = + R( 26)*C(  9)*C( 24) - R( 27)*C(  1)*C( 18) - R( 28)KM  2743
     *    *C(  1)*C( 18)                                                KM  2744
      RATE( 19) = +  0.50*R( 22)*C(  3)*C(  8) - R( 23)*C(  2)*C( 19) - KM  2745
     *    R( 25)*C(  1)*C( 19) - R( 49)*C( 19)*C( 28)                   KM  2746
      RATE( 20) = + R( 11)*C(  2)*C( 11) - R( 21)*C( 20)                KM  2747
      RATE( 21) = + R( 44)*C( 13)*C( 24) - R( 45)*C(  1)*C( 21)         KM  2748
      RATE( 22) = + R( 41)*C( 13)*C( 24) - R( 42)*C( 22)*C( 24)         KM  2749
      RATE( 23) = + R( 19)*C(  1)*C( 17) - R( 20)*C( 23) + R( 51)*C( 17)KM  2750
     *    *C( 28)                                                       KM  2751
      RATE( 24) = + R(  6)*C(  4) - R(  7)*C(  1)*C( 24) - R(  8)*C(  2)KM  2752
     *    *C( 24) - R(  9)*C(  5)*C( 24) + R( 10)*C(  1)*C( 11) - R( 18)KM  2753
     *    *C(  8)*C( 24) +  0.50*R( 22)*C(  3)*C(  8) - R( 26)*C(  9)   KM  2754
     *    *C( 24) - R( 33)*C(  7)*C( 24) - R( 39)*C(  6)*C( 24) - R( 41)KM  2755
     *    *C( 13)*C( 24) - R( 42)*C( 22)*C( 24) - R( 44)*C( 13)*C( 24)  KM  2756
     *     - R( 47)*C( 24)*C( 28) + R( 48)*C( 11)*C( 28)                KM  2757
      RATE( 25) = + R(  1)*C(  2) - R(  2)*C( 25) - R( 24)*C(  8)*C( 25)KM  2758
     *     - R( 46)*C( 25)*C( 28)                                       KM  2759
      RATE( 26) = +  0.85*R( 27)*C(  1)*C( 18) - R( 29)*C( 26) + R( 30) KM  2760
     *    *C(  1)*C( 15) - R( 31)*C( 26) - R( 32)*C(  2)*C( 26)         KM  2761
     *    +R( 50)*C( 25)*C( 28)                                         KM  2762
      RATE( 27) = + R( 15)*C(  2)*C( 10) - R( 16)*C( 27)*C( 30) - R( 17)KM  2763
     *    *C( 27)                                                       KM  2764
      RATE( 28) = - R( 46)*C( 25)*C( 28) - R( 47)*C( 24)*C( 28) - R( 48)KM  2765
     *    *C( 11)*C( 28) - R( 49)*C( 19)*C( 28) - R( 50)*C( 15)*C( 28)  KM  2766
     *     - R( 51)*C( 17)*C( 28)                                       KM  2767
      RATE( 29) = + R( 46)*C( 25)*C( 28) + R( 47)*C( 24)*C( 28) + R( 48)KM  2768
     *    *C( 11)*C( 28) + R( 49)*C( 19)*C( 28) + R( 50)*C( 15)*C( 28)  KM  2769
     *     + R( 51)*C( 17)*C( 28)                                       KM  2770
C     RATE( 30) = - R(  4)*C(  1)*C(  2)*C( 30) + R(  5)*C(  4)** 2 -   KM  2771
C    *    R( 16)*C( 27)*C( 30) + R( 26)*C(  9)*C( 24) + R( 33)*C(  7)   KM  2772
C    *    *C( 24) + R( 39)*C(  6)*C( 24) + R( 44)*C( 13)*C( 24)         KM  2773
C                                                                       KM  2774
      DO 120 J = 1,NK                                                   KM  2775
      YDOT(J+KS) = RATE(J)                                              KM  2776
120   CONTINUE                                                          KM  2777
C                                                                       KM  2778
130   CONTINUE                                                          KM  2779
      RETURN                                                            KM  2780
      END                                                               KM  2781
      SUBROUTINE SCALE(Y1,Y2,YB,YT)
C
C         ESTABLISH MINIMUM AND MAXIMUM VALUES FOR PRINTER-PLOT  Y  AXIS
C
      DIMENSION Y(2)
      DATA ONE,TEN/1.,10./
C
      BOT = Y1
      TOP = Y2
      IF(Y1.LT.Y2) GO TO 10
      TOP = Y1
      BOT = Y2
   10 CONTINUE
      LA = 0
      Y(1) = TOP
      Y(2) = BOT
      Z = TOP
      DO 50 I=1,2
      IF(Z.EQ.0.) GO TO 45
      SN = SIGN(ONE,Z)
      A = ABS(Z)
      LA = IFIX(ALOG10(A))
      IF(SN.LT.0.) GO TO 30
   20 DO 25 N=2,10
      XN = FLOAT(N)
      YT = SN*XN*TEN**LA
      IF(Z.LT.YT) GO TO 40
   25 CONTINUE
      GO TO 40
   30 DO 35 N=1,9
      XN = TEN - FLOAT(N)
      YT = SN*XN*TEN**LA
      IF(Z.LT.YT) GO TO 40
   35 CONTINUE
   40 Y(I) = YT
   45 IF(I.EQ.2) GO TO 50
      Z = BOT
   50 CONTINUE
      YT = Y(1)
      YB = Y(2) - TEN**LA
C
      RETURN
      END


      SUBROUTINE SKEDUL(X,T,NX,NINT,UPDELT,INTIM,TSTOP,DELTIN,FAIL,INTP)
C
C     SUBROUTINE SKEDUL CALCULATES MEAN VALUES FOR A REGULAR UPDATE
C     SCHEDULE  FROM AN IRREGULAR SCHEDULE OR DIFFERENT INTERVAL SCHEDUL
C
C     X IS AN INPUT STEP FUNCTION WITH ARBITARY UPDATE INTERVALS
C     X IS RETURNED AS A STEP FUNCTION WITH A FIXED UPDATE INTERVAL
C     T IS THE TIME SCHEDULE ASSOCIATED WITH X ON INPUT
C       IF T IS AVAILIABLE SET DELTIN EQUAL TO A NEGATIVE NUMBER
```

```
C        IT MUST BE MONOTONICALLY INCREASING
C     NX IS THE NUMBER OF X VALUES INPUT
C  NINT IS THE NUMBER OF UPDATE INTERVALS OF THE RETURNED FUNCTION X
C UPDELT IS THE DESIRED UPDATE INTERVAL INPUT
C INTIM IS THE INITIAL TIME TO START UPDATING
C TSTOP IS THE FINAL TIME
C DELTIN IS A FIXED UPDATE INTERVAL OF X INPUT (DELTIN .NE. UPDELT
C          IS USED WHEN T IS NOT AVAILIABLE
C  FAIL IS RETURN AS 'YES' IF T IS NOT MONOTONICALLY INCREASING
C  INTP IS A FLAG POSITIVE FOR INTERPOLATION, NEGATIVE FOR STEP-WISE
C               INTEGRATION
C
      REAL INTIM
      DIMENSION X(1), T(1), XO(200), TO(200)
      DATA  YES  /3HYES/
      DATA  RNEG  /2HNO/
      FAIL = RNEG
      IF(INTP.GT.0)   GO TO 330
C     INITIALIZE T IF NOT INPUT
      IF(DELTIN.LT.0.0)  GO TO 10
      T(1) = INTIM
      DO 5 I=2,NX
      T(I) = T(I-1) + DELTIN
    5 CONTINUE
C     INITIALIZE TIME OUT SCHEDULE
   10 NINT = (TSTOP+ .01 -INTIM)/UPDELT + 1
      TO(1) = INTIM
      NP1 = NINT + 1
      TO(2) =INTIM + UPDELT/2.
      DO 15 I=3,NP1
      TO(I) = TO(I-1) + UPDELT
   15 CONTINUE
C     CHECK FOR FIRST T
      IF(T(1) .GE. TO(1))  GO TO 60
      DO 50 J=2,NX
      IF(T(J) .LT. TO(1))  GO TO 50
      KL = J-1
      GO TO 70
   50 CONTINUE
      K = NX +1
      GO TO 320
   60 KL=1
   70 T(KL) = TO(1)
      T(NX+1) = TO(NP1) + .01
      IF(INTP.GT.0)  GO TO 330
      S=0.0
C     FORWARD INTEGRATIOON LOOP TO DETERMINE MEAN VALUES OF NEW INTERVAL
      DO 300 I=1,NINT
      IF(I.EQ.1)  GO TO 80
      IF(T(KL).GE.TO(I+1))  GO TO 280
      IF(KL.GT.1)  S=X(KL-1)*(T(KL)- TO(I))
   80 DO 200 K=KL,NX
      IF(T(K+1).LT.T(K))  GO TO 320
      IF(T(K+1).GE.TO(I+1))  GO TO 160
      S = S + X(K)*(T(K+1)-T(K))
      GO TO 200
  160 S = S + X(K)*(TO(I+1)-T(K))
      GO TO 250
  200 CONTINUE
C
  250 XO(I) = S/UPDELT
      KL = K + 1
      GO TO 300
  280 XO(I) = X(K)
  300 CONTINUE
C
      CALL XMIT(NINT,XO,X)
      RETURN
C
  330 CONTINUE
      SMALL = 1.E-3
      TO(1) = INTIM
      NINT = IFIX((TSTOP - INTIM + SMALL)/UPDELT) + 1
      NP1 = NINT + 1
      KL = 1
      KLP = 2
      DO 340 I = 2,NP1
      TO(I) = TO(I-1) + UPDELT
340   CONTINUE
      X(NX+1) = X(NX)
      DO 350 I = 1,NINT
      TMX = T(KLP) - SMALL
      IF(TO(I).GT.TMX)  KL = KL + 1
      KLP = KL + 1
      SLOPE = (X(KLP)-X(KL)) / (T(KLP)-T(KL))
      XO(I) = X(KL) + SLOPE*(TO(I)-T(KL))
  350 CONTINUE
      CALL XMIT(NINT,XO,X)
      RETURN
C
  320 WRITE(6,1) T(K+1)
  1   FORMAT(1H1,34HRESCHEDULING FAILED AT TIME EQUAL  ,F8.2)
      FAIL = YES
      RETURN
      END


      SUBROUTINE SOL (N, NDIM, A, B, IP)
C-----------------------------------------------------------------------
C SOLUTION OF LINEAR SYSTEM, A*X = B .
C INPUT..
C    N = ORDER OF MATRIX.
C    NDIM = DECLARED DIMENSION OF ARRAY  A .
C    A = TRIANGULARIZED MATRIX OBTAINED FROM DEC.
C    B = RIGHT HAND SIDE VECTOR.
C    IP = PIVOT VECTOR OBTAINED FROM DEC.
C DO NOT USE IF DEC HAS SET IER .NE. 0.
C OUTPUT..
C    B = SOLUTION VECTOR, X .
```

```
C-----------------------------------------------------------------------
      INTEGER IP, N, NDIM
      INTEGER I, K, KB, KM1, KP1, M, NM1
      DIMENSION A(NDIM, N), B(N), IP(N)
C
      IF (N .EQ. 1) GO TO 50
      NM1 = N - 1
      DO 20 K = 1,NM1
        KP1 = K + 1
        M = IP(K)
        T = B(M)
        B(M) = B(K)
        B(K) = T
        DO 10 I = KP1,N
 10       B(I) = B(I) + A(I,K)*T
 20     CONTINUE
      DO 40 KB = 1,NM1
        KM1 = N - KB
        K = KM1 + 1
        B(K) = B(K)/A(K,K)
        T = -B(K)
        DO 30 I = 1,KM1
 30       B(I) = B(I) + A(I,K)*T
 40     CONTINUE
 50   B(1) = B(1)/A(1,1)
      RETURN
C----------------------- END OF SUBROUTINE SOL --------------------------
      END


      SUBROUTINE SOL2B2(A,B)
C     THIS SUBROUTINE SOLVES A SYSTEM OF 2 LINEAR EQUATIONS
      DIMENSION A(2,2), B(2), X(2)
      DET = A(1,1)*A(2,2) - A(1,2)*A(2,1)
      X(1) = ( A(2,2)*B(1) - A(1,2)*B(2) )/DET
      X(2) = (-A(2,1)*B(1) + A(1,1)*B(2) )/DET
      B(1) = X(1)
      B(2) = X(2)
      RETURN
      END


      SUBROUTINE STEADY (Y,N)
C
C         H2O IS TREATED AT A CONSTANT USING NSTDY = 1
C         THERE ARE NO STEADY STATE APPROXIMATIONS INCLUDED
C         IN THIS VERSION OF STEADY.
C
      RETURN
      END
      SUBROUTINE TEMPR(IT,TIME,T)
C
C     DETERMINES TEMPERATURE DEPENDENT REACTION RATES
C
      DIMENSION T(1)
      COMMON/CHEM2/ CONIN(40,5),       WTMOLE(40),      RATKON(55),
     1              RATEFF(55),        RATEV(2,5),      QRATE,
     2              NVRATE,            LOCVRT(2)
      RT = 1.986*T(IT)
C     HNO3 DISSOCIATION RATE
      RATKON(21) = 7.8E15 * EXP(-20700./RT)
      RATEFF(21) = RATKON(21)
C     PAN DISSOCIATION RATE
      RATKON(37) = 1.17E17 * EXP(-26910./RT)
      RATEFF(37) = RATKON(37)
      WRITE(10,1)  TIME, T(IT), RATKON(21), RATKON(37)
      RETURN
    1 FORMAT(1H010X,40HTEMPERATURE DEPENDENT RATES' TIME =          ,
     1  ,F8.2,5X,6HTEMP =,F8.2,5X,7HRATES = ,2E13.3)
      END


      SUBROUTINE TIMEX
      RETURN
      END


      SUBROUTINE TSTEP (Y, NO)
C
C-----------------------------------------------------------------------
C     THIS VERSION OF TSTEP SOLVES A BLOCK TRI-DIAGONAL SYSTEM OF ODE*S
C     AND    IS DESIGNED FOR MF = 21 ONLY
C     MODIFICATIONS BY F.W. LURMANN   (7.8.77).
C-----------------------------------------------------------------------
C TSTEP PERFORMS ONE STEP OF THE INTEGRATION OF AN INITIAL VALUE
C PROBLEM FOR A SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS.
C-----------------------------------------------------------------------
      INTEGER NO
      INTEGER IPIV, JSTART, KFLAG, L, LMAX, METH, MF, N, NFE, NJE,
     1        NQ, NQINDX, NQUSED, NSTEP
      INTEGER I, IBACK, IER, IREDO, J, J1, J2, M, MFOLD, MIO,
     1        MITER, MITER1, NEWJ, NSTEPJ
      INTEGER ISTEPJ, KFC, KFH, MAXCOR
C/24
      DIMENSION Y(NO, 6)
C
      COMMON /EPCOM1/ T,H,HMIN,HMAX,EPS,SS,UROUND,N,MF,KFLAG,JSTART
      COMMON /EPCOM2/ YMAX(1)
      COMMON /EPCOM3/ ERROR(1)
      COMMON /EPCOM4/ SAVE1(1)
      COMMON /EPCOM5/ SAVE2(1)
      COMMON /EPCOM6/ PW(1)
      COMMON /EPCOM7/ IPIV(1)
      COMMON /EPCOM9/ HUSED,NQUSED,NSTEP,NFE,NJE                        KM  3036
      COMMON /EPCM10/ TAU(13),EL(13),TQ(5),LMAX,METH,NQ,L,NQINDX        KM  3037
      COMMON /EPCM12/ PWL(3600),  WORKR(30), DUPPER(4), DLOWER(4)       KM  3038
      INTEGER BCFLAG                                                    KM  3039
      COMMON/CHEM1/ NOSTAT,              NOSTM1,             NOREAC,    KM  3040
     1              NOSPEC,              NSTDY,              NK         KM  3041
      COMMON/CHEM2/ CONIN(40,5),         WTMOLE(40),         RATKON(55), KM  3042
     1              RATEFF(55),          RATEV(2,5),         QRATE,     KM  3043
     2              NVRATE,              LOCVRT(2)                      KM  3044
      COMMON/CHEM3/ ZEE(5),              DELZ(4),            HTCELL(6), KM  3045
     1              TDELZ(2) ,           DFINIT(6),          SCALOW(4), KM  3046
     2              DCOF(5),             FLXWAL(40),         FLXDGE(40), KM  3047
     3              BCFLAG(40),          DPRATE(10),         DEPOWR(10), KM  3048
     4              LOCDPF(10),          NDPFLX ,            SCALUP(4)  KM  3049
C                                                                       KM  3050
      DATA ISTEPJ /20/, KFC /-3/, KFH /-7/, MAXCOR /3/                  KM  3051
C(6                                                                     KM  3052
      DATA ADDON  /1.0E-6/,    BIAS1  /2.5E1/, BIAS2  /2.5E1/,          KM  3053
     1     BIAS3  /1.0E2/,     CROOWN /0.1E0/, DELRC  /0.3E0/,          KM  3054
     2     ETACF  /0.25E0/,    ETAMIN /0.1E0/, ETAMXF /0.2E0/,          KM  3055
     3     ETAMX1 /1.0E4/,     ETAMX2 /1.0E1/, ETAMX3 /1.5E0/,          KM  3056
     4     ONEPSM /1.00001E0/, SHORT  /0.1E0/, THRESH /1.3E0/           KM  3057
      DATA ONE /1.0E0/, PT5 /0.5E0/, ZERO /0.0E0/                       KM  3058
      KFLAG = 0                                                         KM  3059
      TOLD = T                                                          KM  3060
      FLOTN =  FLOAT(N)                                                 KM  3061
      IF (JSTART .GT. 0) GO TO 200                                      KM  3062
      IF (JSTART .NE. 0) GO TO 150                                      KM  3063
C-----------------------------------------------------------------------KM  3064
C ON THE FIRST CALL, THE ORDER IS SET TO 1 AND THE INITIAL              KM  3065
C DERIVATIVES ARE CALCULATED.  ETAMAX IS THE MAXIMUM RATIO BY           KM  3066
C WHICH H CAN BE INCREASED IN A SINGLE STEP.  IT IS 1.E04 FOR THE       KM  3067
C FIRST STEP TO COMPENSATE FOR THE SMALL INITIAL H, THEN 10 FOR         KM  3068
C THE NEXT 10 STEPS, AND THEN 1.5 THEREAFTER.  IF A FAILURE             KM  3069
C OCCURS (IN CORRECTOR CONVERGENCE OR ERROR TEST), ETAMAX IS SET AT 1   KM  3070
C FOR THE NEXT INCREASE.  ETAMIN = .1 IS THE MINIMUM RATIO BY WHICH     KM  3071
C H CAN BE REDUCED ON ANY RETRY OF A STEP.                              KM  3072
C-----------------------------------------------------------------------KM  3073
      CALL DIFFUN (N, T, Y, SAVE1)                                      KM  3074
      DO 110 I = 1,N                                                    KM  3075
 110    Y(I,2) = H*SAVE1(I)                                             KM  3076
      METH = MF/10                                                      KM  3077
      MITER = MF - 10*METH                                              KM  3078
      MITER1 = MITER + 1                                                KM  3079
      MFOLD = MF                                                        KM  3080
      NQ = 1                                                            KM  3081
      L = 2                                                             KM  3082
      TAU(1) = H                                                        KM  3083
      PRL1 = ONE                                                        KM  3084
      RC = ZERO                                                         KM  3085
      ETAMAX = ETAMX1                                                   KM  3086
      NQINDX = 2                                                        KM  3087
      NSTEP = 0                                                         KM  3088
      NSTEPJ = 0                                                        KM  3089
      NFE = 1                                                           KM  3090
      NJE = 0
      GO TO 200
C-----------------------------------------------------------------------
C IF THE USER HAS CHANGED H, THEN Y MUST BE RESCALED.  IF THE
C USER HAS CHANGED MITER, THEN NEWJ IS SET TO MITER TO FORCE
C THE PARTIAL DERIVATIVEES TO BE UPDATED, IF THEY ARE BEING USED.
C-----------------------------------------------------------------------
 150  IF (MF .EQ. MFOLD) GO TO 170
      MIO = MITER
      METH = MF/10
      MITER = MF - 10*METH
      MFOLD = MF
      IF (MITER .EQ. MIO) GO TO 170
      NEWJ = MITER
      MITER1 = MITER + 1
 170  IF (H .EQ. HOLD) GO TO 200
      ETA = H/HOLD
      H = HOLD
      IREDO = 3
      GO TO 185
C(2
  180 ETA = AMAX1(ETA,HMIN/ ABS(H),ETAMIN)
  185 ETA = AMIN1(ETA,HMAX/ ABS(H),ETAMAX)
      R1 = ONE
      DO 190 J = 2,L
        R1 = R1*ETA
        DO 190 I = 1,N
 190      Y(I,J) = Y(I,J)*R1
      H = H*ETA
      RC = RC*ETA
      IF (IREDO .EQ. 0) GO TO 690
C-----------------------------------------------------------------------
C THIS SECTION COMPUTES THE PREDICTED VALUES BY EFFECTIVELY
C MULTIPLYING THE Y ARRAY BY THE PASCAL TRIANGLE MATRIX.  THEN
C COSET IS CALLED TO OBTAIN EL, THE VECTOR OF COEFFICIENTS OF
C LENGTH NQ + 1.  RC IS THE RATIO OF NEW TO OLD VALUES OF THE
C COEFFICIENT H/EL(2).  WHEN RC DIFFERS FROM 1 BY MORE THAN
C DELRC, NEWJ IS SET TO MITER TO FORCE THE PARTIAL DERIVATIVES
C TO BE UPDATED, IF USED.  DELRC IS 0.3.  IN ANY CASE, THE PARTIAL
C DERIVATIVES ARE UPDATED AT LEAST EVERY 20-TH STEP.
C-----------------------------------------------------------------------
 200  T = T + H
      DO 210 J1 = 1,NQ
        DO 210 J2 = J1,NQ
          J = (NQ + J1) - J2
          DO 210 I = 1,N
 210        Y(I,J) = Y(I,J) + Y(I,J+1)
      CALL COSET
      BND = FLOTN*(TQ(4)*EPS)**2
      RL1 = ONE/EL(2)
      RC = RC*(RL1/PRL1)
      PRL1 = RL1
      IF (NSTEP .GE. NSTEPJ+ISTEPJ) NEWJ = MITER
C(1
      DRC =  ABS(RC-ONE)
      IF (DRC .LE. DELRC) GO TO 215
      NEWJ = MITER
      CRATE = ONE
      RC = ONE
      GO TO 220
  215 IF ((MITER .NE. 0) .AND. (DRC .NE. ZERO)) CRATE = ONE
C-----------------------------------------------------------------------
C UP TO 3 CORRECTOR ITERATIONS ARE TAKEN.  A CONVERGENCE TEST IS MADE
C ON THE ROOT MEAN SQUARE NORM OF EACH CORRECTION, USING BND, WHICH
C IS DEPENDENT ON EPS.  THE SUM OF THE CORRECTIONS IS ACCUMULATED IN
C THE VECTOR ERROR.  THE Y ARRAY IS NOT ALTERED IN THE CORRECTOR
C LOOP.  THE UPDATED Y VECTOR IS STORED TEMPORARILY IN SAVE1.
C-----------------------------------------------------------------------
 220  DO 230 I = 1,N
 230    ERROR(I) = ZERO
      M = 0
      CALL DIFFUN (N, T, Y, SAVE2)
      NFE = NFE + 1
      IF (NEWJ .LE. 0) GO TO 290
C-----------------------------------------------------------------------
C IF INDICATED, THE MATRIX P = I - H*RL1*J IS REEVALUATED BEFORE
C STARTING THE CORRECTOR ITERATION.  NEWJ IS SET TO 0 AS AN
C INDICATOR THAT THIS HAS BEEN DONE.  IF MITER = 1 OR 2, P IS
C COMPUTED AND PROCESSED IN PSET.  IF MITER = 3, THE MATRIX  IS
C P = I - H*RL1*D, WHERE D IS A DIAGONAL MATRIX.  RL1 IS 1/EL(2).
C-----------------------------------------------------------------------
      NEWJ = 0
      RC = ONE
      NJE = NJE + 1
      NSTEPJ = NSTEP
      GO TO (250, 240, 260), MITER
 240  NFE = NFE + N
 250  CON = -H*RL1
      CALL PSET(Y, N0, CON, MITER, IER)
      IF (IER .NE. 0) GO TO 420
      GO TO 350
 260  R = RL1*SHORT
      DO 270 I = 1,N
 270    PW(I) = Y(I,1) + R*(H*SAVE2(I) - Y(I,2))
      CALL DIFFUN(N, T, PW, SAVE1)
      NFE = NFE + 1
      HRL1 = H*RL1
      DO 280 I = 1,N
        R0 = H*SAVE2(I) - Y(I,2)
        PW(I) = ONE
        D = SHORT*R0 - H*(SAVE1(I) - SAVE2(I))
        SAVE1(I) = ZERO
C(2
        IF ( ABS(R0) .LT. UROUND*YMAX(I)) GO TO 280
        IF ( ABS(D) .EQ. ZERO) GO TO 420
        PW(I) = SHORT*R0/D
        SAVE1(I) = PW(I)*RL1*R0
 280    CONTINUE
      GO TO 370
 290  GO TO (295, 350, 350, 310), MITER1
```

```
C-----------------------------------------------------------------------
C IN THE CASE OF FUNCTIONAL ITERATION, Y IS UPDATED DIRECTLY FROM
C THE RESULT OF THE LAST DIFFUN CALL.
C-----------------------------------------------------------------------
 295  D = ZERO
      DO 300 I = 1,N
        R = RL1*(H*SAVE2(I) - Y(I,2))
        D = D + ((R - ERROR(I))/YMAX(I))**2
        SAVE1(I) = Y(I,1) + R
 300    ERROR(I) = R
      GO TO 400
C-----------------------------------------------------------------------
C IN THE CASE OF A CHORD METHOD, THE RESIDUAL -G(Y SUB N(M))
C IS COMPUTED AND THE LINEAR SYSTEM WITH THAT AS RIGHT-HAND SIDE
C AND P AS COEFFICIENT MATRIX IS SOLVED, USING THE LU DECOMPOSITION
C OF P IF MITER = 1 OR 2.  IF MITER = 3 THE SCALAR H*RL1 IS UPDATED.
C-----------------------------------------------------------------------
 310  PHRL1 = HRL1
      HRL1 = H*RL1
      IF (HRL1 .EQ. PHRL1) GO TO 330
      R = HRL1/PHRL1
      DO 320 I = 1,N
        D = ONE - R*(ONE - ONE/PW(I))
C(1
        IF ( ABS(D) .EQ. ZERO) GO TO 440
C)1
CS      IF (ABS(D) .EQ. ZERO) GO TO 440
C/4
 320    PW(I) = ONE/D
 330  DO 340 I = 1,N
 340    SAVE1(I) = PW(I)*(RL1*H*SAVE2(I) - (RL1*Y(I,2) + ERROR(I)))
      GO TO 370
 350  DO 360 I = 1,N
 360    SAVE1(I) = RL1*H*SAVE2(I) - (RL1*Y(I,2) + ERROR(I))
      CALL BLKSOL(PW,NK,NOSTAT,SAVE1,IPIV,DLOWER,DUPPER,PWL,WORKR,BCFLAG
     1)
CERT    CALL SOL (N,  NO,  PW,  SAVE1,  IPIV)
 370  D = ZERO
      DO 380 I = 1,N
        ERROR(I) = ERROR(I) + SAVE1(I)
        D = D + (SAVE1(I)/YMAX(I))**2
 380    SAVE1(I) = Y(I,1) + ERROR(I)
C-----------------------------------------------------------------------
C TEST FOR CONVERGENCE.  IF M .GT. 0, AN ESTIMATE OF THE SQUARE OF
C THE CONVERGENCE RATE CONSTANT IS STORED IN CRATE, AND THIS IS USED
C IN THE TEST.
C-----------------------------------------------------------------------
C(2
 400  IF (M .NE. 0) CRATE = AMAX1(CRDOWN*CRATE,D/D1)
      IF (D*AMIN1(ONE,CRATE) .LE. BND) GO TO 450
      D1 = D
      M = M + 1
      IF (M .EQ. MAXCOR) GO TO 410
      CALL DIFFUN (N, T, SAVE1, SAVE2)
      GO TO (295, 350, 350, 310), MITER1
```

```
C-------------------------------------------------------------------
C THE CORRECTOR ITERATION FAILED TO CONVERGE IN 3 TRIES. IF PARTIAL
C DERIVATIVES ARE INVOLVED BUT ARE NOT UP TO DATE, THEY ARE
C REEVALUATED FOR THE NEXT TRY.  OTHERWISE THE Y ARRAY IS RESTORED
C TO ITS VALUES BEFORE PREDICTION, AND H IS REDUCED,
C IF POSSIBLE.  IF NOT, A NO-CONVERGENCE EXIT IS TAKEN.
C-------------------------------------------------------------------
 410  NFE = NFE + MAXCOR - 1
      IF (NEWJ .EQ. -1) GO TO 440
 420  T = TOLD
      ETAMAX = ONE
      DO 430 J1 = 1,NQ
        DO 430 J2 = J1,NQ
          J = (NQ + J1) - J2
          DO 430 I = 1,N
 430        Y(I,J) = Y(I,J) - Y(I,J+1)
C(1
      IF ( ABS(H) .LE. HMIN*ONEPSM) GO TO 680
      ETA = ETACF
      IREDO = 1
      GO TO 180
 440  NEWJ = MITER
      GO TO 220
C-------------------------------------------------------------------
C THE CORRECTOR HAS CONVERGED.  NEWJ IS SET TO -1 IF PARTIAL
C DERIVATIVES WERE USED, TO SIGNAL THAT THEY MAY NEED UPDATING ON
C SUBSEQUENT STEPS.  THE ERROR TEST IS MADE AND CONTROL PASSES TO
C STATEMENT 500 IF IT FAILS.
C-------------------------------------------------------------------
 450  IF (MITER .NE. 0) NEWJ = -1
      NFE = NFE + M
      D = ZERO
      DO 460 I = 1,N
 460    D = D + (ERROR(I)/YMAX(I))**2
      E = FLOTN*(TQ(2)*EPS)**2
      IF (D .GT. E) GO TO 500
C-------------------------------------------------------------------
C AFTER A SUCCESSFUL STEP, THE Y ARRAY, TAU, NSTEP, AND NQINDX ARE
C UPDATED, AND A NEW VALUE OF H AT ORDER NQ IS COMPUTED.
C THE VECTOR TAU CONTAINS THE NQ + 1 MOST RECENT VALUES OF H.
C A CHANGE IN NQ UP OR DOWN BY 1 IS CONSIDERED IF NQINDX = 0.
C IF NQINDX = 1 AND NQ .LT. MAXDER, THEN ERROR IS SAVED
C FOR USE IN A POSSIBLE ORDER INCREASE ON THE NEXT STEP.
C A CHANGE IN H OR NQ IS MADE ONLY OF THE INCREASE IN H
C IS BY A FACTOR OF AT LEAST 1.3.
C IF NOT, NQINDX IS SET TO 2 TO PREVENT TESTING FOR THAT MANY
C STEPS.  IF NQ IS CHANGED, NQINDX IS SET TO NQ + 1 (NEW VALUE).
C-------------------------------------------------------------------
      KFLAG = 0
      IREDO = 0
      NSTEP = NSTEP + 1
      HUSED = H
      NQUSED = NQ
      DO 470 IBACK = 1,NQ
        I = L - IBACK
 470      TAU(I+1) = TAU(I)
        TAU(1) = H
        DO 480 J = 1,L
          DO 480 I = 1,N
 480        Y(I,J) = Y(I,J) + ERROR(I)*EL(J)
        NQINDX = NQINDX - 1
        IF ((L .EQ. LMAX) .OR. (NQINDX .NE. 1)) GO TO 495
        DO 490 I = 1,N
 490      Y(I,LMAX) = ERROR(I)
        CONP = TQ(5)
 495    IF (ETAMAX .NE. ONE) GO TO 520
        IF (NQINDX .LT. 2) NQINDX = 2
        GO TO 690
C-----------------------------------------------------------------------
C THE ERROR TEST FAILED.  KFLAG KEEPS TRACK OF MULTIPLE FAILURES.
C T AND THE Y ARRAY ARE RESTORED TO THEIR PREVIOUS VALUES.  A NEW
C H FOR A RETRY OF THE STEP IS COMPUTED.  THE ORDER IS KEPT FIXED.
C-----------------------------------------------------------------------
 500    KFLAG = KFLAG - 1
        T = TOLD
        DO 510 J1 = 1,NQ
          DO 510 J2 = J1,NQ
            J = (NQ + J1) - J2
            DO 510 I = 1,N
 510          Y(I,J) = Y(I,J) - Y(I,J+1)
        NEWJ = MITER
        ETAMAX = ONE
C(1
        IF ( ABS(H) .LE. HMIN*ONEPSM) GO TO 660
        IF (KFLAG .LE. KFC) GO TO 630
        IREDO = 2
C COMPUTE RATIO OF NEW H TO CURRENT H AT THE CURRENT ORDER. ------------
C(1
  520 FLOTL =  FLOAT(L)
        ETAQ = ONE/((BIAS2*D/E)**(PT5/FLOTL) + ADDON)
        IF ((NQINDX .NE. 0) .OR. (KFLAG .NE. 0)) GO TO 580
        ETAQM1 = ZERO
        IF (NQ .EQ. 1) GO TO 540
C COMPUTE RATIO OF NEW H TO CURRENT H AT THE CURRENT ORDER LESS ONE. ---
        D = ZERO
        DO 530 I = 1,N
 530      D = D + (Y(I,L)/YMAX(I))**2
        EDN = FLOTN*(TQ(1)*EPS)**2
        ETAQM1 = ONE/((BIAS1*D/EDN)**(PT5/(FLOTL - ONE)) + ADDON)
 540    ETAQP1 = ZERO
        IF (L .EQ. LMAX) GO TO 560
C COMPUTE RATIO OF NEW H TO CURRENT H AT CURRENT ORDER PLUS ONE. -------
        CNQUOT = (TQ(5)/CONP)*(H/TAU(2))**L
        D = ZERO
        DO 550 I = 1,N
 550      D = D + ((ERROR(I) - CNQUOT*Y(I,LMAX))/YMAX(I))**2
        EUP = FLOTN*(TQ(3)*EPS)**2
        ETAQP1 = ONE/((BIAS3*D/EUP)**(PT5/(FLOTL + ONE)) + ADDON)
 560    NQINDX = 2
        IF (ETAQ .GE. ETAQP1) GO TO 570
```

```
      IF (ETAQP1 .GT. ETAQM1) GO TO 600
      GO TO 590
 570  IF (ETAQ .LT. ETAQM1) GO TO 590
 580  IF ((ETAQ .LT. THRESH) .AND. (KFLAG .EQ. 0)) GO TO 690
      ETA = ETAQ
      IF ((KFLAG .LE. -2) .AND. (ETA .GT. ETAMXF)) ETA = ETAMXF
      GO TO 180
 590  IF (ETAQM1 .LT. THRESH) GO TO 690
      CALL ADJUST (Y, NQ)
      L = NQ
      NQ = NQ - 1
      ETA = ETAQM1
      NQINDX = L
      GO TO 180
 600  IF (ETAQP1 .LT. THRESH) GO TO 690
      NQ = L
      ETA = ETAQP1
      L = L + 1
      DO 610 I = 1,N
 610    Y(I,L) = ZERO
      NQINDX = L
      GO TO 180
C-----------------------------------------------------------------------
C CONTROL REACHES THIS SECTION IF 3 OR MORE CONSECUTIVE FAILURES
C HAVE OCCURRED.  IT IS ASSUMED THAT THE ELEMENTS OF THE Y ARRAY
C HAVE ACCUMULATED ERRORS OF THE WRONG ORDER.  THE ORDER IS REDUCED
C BY ONE, IF POSSIBLE.  THEN H IS REDUCED BY A FACTOR OF 0.1 AND
C THE STEP IS RETRIED.  AFTER A TOTAL OF 7 CONSECUTIVE FAILURES,
C AN EXIT IS TAKEN WITH KFLAG = -2.
C-----------------------------------------------------------------------
 630  IF (KFLAG .EQ. KFH) GO TO 670
      IF (NQ .EQ. 1) GO TO 640
      ETA = ETAMIN
      CALL ADJUST (Y, NQ)
      L = NQ
      NQ = NQ - 1
      NQINDX = L
      GO TO 180
C(1
  640 ETA = AMAX1(ETAMIN,HMIN/ ABS(H))
      H = H*ETA
      CALL DIFFUN (N, T, Y, SAVE1)
      NFE = NFE + 1
      DO 650 I = 1,N
 650    Y(I,2) = H*SAVE1(I)
      NQINDX = 10
      GO TO 200
C-----------------------------------------------------------------------
C ALL RETURNS ARE MADE THROUGH THIS SECTION.  H IS SAVED IN HOLD
C TO ALLOW THE CALLER TO CHANGE H ON THE NEXT STEP.
C-----------------------------------------------------------------------
 660  KFLAG = -1
      GO TO 700
 670  KFLAG = -2
      GO TO 700
  680 KFLAG = -3
      GO TO 700
  690 ETAMAX = ETAMX3
      IF (NSTEP .LE. 10) ETAMAX = ETAMX2
  700 HOLD = H
      JSTART = NQ
      RETURN
C----------------------- END OF SUBROUTINE TSTEP --------------------
      END



      SUBROUTINE UNMIXR
      RETURN
      END



      SUBROUTINE UPFLX1(TIME,J)
C
C        *UPFLX1* UPDATES AREA SOURCE EMISSION FLUXES
C
C         THIS ROUTINE HAS BEEN MODIFIED TO PARTITION A SINGLE
C         ALDEHYDE FLUX INTO 60% HCHO AND 40% RCHO.
C
      COMMON/FLUXES/FLXIN( 7,200),  FLXTIM(200),   NFLUX
      COMMON/CHEM3/ ZEE(5),                DELZ(4),           HTCELL(6),
     1              TDELZ(2) ,             DFINIT(6),         SCALOW(4),
     2              DCOF(5),               FLXWAL(40),        FLXDGE(40),
     3              BCFLAG(40),            DPRATE(10),        DEPOWR(10),
     4              LOCDPF(10),            NDPFLX ,           SCALUP(4)
      COMMON /CHEM5/  REACT(20,55),SPEC(40),LOCFLX(10),
     1                NASFLX,FLXW1(30),FLXW2(30)
      INTEGER BCFLAG
      IF(NASFLX.LE.0)  RETURN
      WRITE(10,1)TIME
      JP1 =MIN0(J+1,NFLUX)
      DO 100 I = 1,NASFLX
      K = LOCFLX(I)
      FLXW1(K) = FLXIN(I,J)
      FLXW2(K) = FLXIN(I,JP1)
  100 CONTINUE
C
C         SPLIT ALDEHYDES
C
      FLXW1(7) = 0.40*FLXW1(6)
      FLXW2(7) = 0.40*FLXW2(6)
      FLXW1(6) = 0.60*FLXW1(6)
      FLXW2(6) = 0.60*FLXW2(6)
C
C       SPLIT NOX
C
      FLXW1(2) = 0.10*FLXW1(1)
      FLXW2(2) = 0.10*FLXW2(1)
      FLXW1(1) = 0.90*FLXW1(1)
      FLXW2(1) = 0.90*FLXW2(1)
      DO 200 I = 1,NASFLX
      K = LOCFLX(I)
      WRITE(10,2) SPEC(K),FLXW1(K)
200   CONTINUE
      K = 7
      WRITE(10,2) SPEC(K),FLXW1(K)
      K = 2
      WRITE(10,2) SPEC(K),FLXW1(K)
1     FORMAT(1H0,10X,38HAREA SOURCE FLUXES UPDATED AT TIME    ,F10.2)
2     FORMAT(1H ,10X,A4,3H = ,2E15.5)
      RETURN
      END


      SUBROUTINE UPRAT2 (T,IK1,RATEV,NOSTAT,NVRATE,CLOUDY)
C
C     -------- UPDATES THE ARRAY OF VARIABLE RATE CONSTANTS (RATEV)
C              FROM RATK1 + RATK2
C     -------- IT INITIALIZES UPPER ELEVATION RATE CONSTANTS  TO
C              THE RATE CONSTANT AT THE SURFACE IF HIRATE = NO
C
      DIMENSION   RATEV(2,5)
      COMMON/CHEM4/RATK1(100,5),        RATK2(100,5),       RLAT,
     1             RLONG,               TMZONE,             SUNTIM,
     2             HIRATE,              JDATE,              NRATE
      DATA YES,NEG,MORE,END/3HYES,2HNO,4HMORE,3HEND/
      IF(HIRATE.EQ.YES)  GO TO 5
      DO 3 J=2,NOSTAT
      RATK1(IK1,J) = RATK1(IK1,1)
      IF(NVRATE.LT.2)  GO TO 3
      RATK2(IK1,J) = RATK2(IK1,1)
    3 CONTINUE
C
    5 DO 10 J=1,NOSTAT
      RATEV(1,J) = RATK1(IK1,J)*CLOUDY
      IF(NVRATE.LT.2)  GO TO 10
      RATEV(2,J) = RATK2(IK1,J)*CLOUDY
   10 CONTINUE
      WRITE(10,20) T  ,   ((RATEV(I,J),J=1,NOSTAT),I =1,NVRATE)
   20 FORMAT (1H0,10X,41HVARIABLE RATE CONSTANTS UPDATED AT TIME ,F6.1,
     . /,11X,3HK1=,7X,5G12.4,/,11X,8HK(HCHO)=,2X,5G12.4)
      RETURN
      END
      SUBROUTINE UPSORC(TIME,IPS,NOSTAT,SPEC)
C
C         UPDATES POINT SOURCE EMISSION RATES
C
      DIMENSION SPEC(1)
      COMMON /PS1/   IPASS(200),      PS(7,5,75),      FRACT(3),
     1               NPTSR,           NPSFLX,          LOCPSF(7)
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT
C
      IF(NPSFLX.LE.0)  GO TO 5
      IF(IPS.LE.NPTSR)  GO TO 10
    5 CALL XMIT(-150,0.0,PSRATE)
      RETURN
   10 CONTINUE
      IPSP1 = MIN0(IPS+1,NPTSR)
      DO 20 J = 1,NOSTAT
      DO 20 I = 1,NPSFLX
      K = LOCPSF(I)
      PSR1(K,J) = PS(I,J,IPS)
      PSR2(K,J) = PS(I,J,IPSP1)
   20 CONTINUE
C
C        SPLIT ALDEHYDES AND NOX
C
      DO 30 J = 1,NOSTAT
      PSR1(7,J) = 0.40*PSR1(6,J)
      PSR2(7,J) = 0.40*PSR2(6,J)
      PSR1(6,J) = 0.60*PSR1(6,J)
      PSR2(6,J) = 0.60*PSR2(6,J)
      PSR1(2,J) = 0.10*PSR1(1,J)
      PSR2(2,J) = 0.10*PSR2(1,J)
      PSR1(1,J) = 0.90*PSR1(1,J)
      PSR2(1,J) = 0.90*PSR2(1,J)
   30 CONTINUE
C
C        WRITE UPDATES ON TAPE10
C
      WRITE(10,1)  TIME
      DO 40 K = 1,NPSFLX
      ID = LOCPSF(K)
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)
   40 CONTINUE
      ID = 2
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)
      ID = 7
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)
      RETURN
    1 FORMAT(1H0,10X,37HPOINT SOURCE RATES UPDATED AT TIME =  ,F8.2)
    2 FORMAT(1H ,10X,A4,4H =    ,5E12.3)
      END
```

4. Utility Library Listing

```
      SUBROUTINE FMINF(F,NF,FMIN,NMIN)
C
C        FMINF LOCATES THE MINIMUM VALUE AMONG NF CONSECUTIVE
C        MEMBERS OF THE F ARRAY AND RETURNS BOTH THE VALUE FMIN
C        AND ITS ARRAY INDEX NMIN.
C
      DIMENSION F(NF)
C
      A = F(1)
      DO 100 N=2,NF
      A = AMIN1(F(N),A)
  100 CONTINUE
      DO 110 N=1,NF
      IF(A-F(N)) 110,120,120
  110 CONTINUE
      N = NF
  120 NMIN = N
      FMIN = A
      RETURN
C
      END
```

```
      SUBROUTINE MCHAR (IFC,FROM,ITC,TO,NCHR)
C
C     MCHAR MOVES A STRING OF CHARACTERS FROM ONE WORD TO ANOTHER.
C
C      IFC  = POSITION OF CHARACTER TO MOVE
C      FROM = SOURCE WORD
C      ITC  =  POSITION OF CHARACTER TO MOVE TO
C      TO   = DESTINATION WORD
C      NCHR = NUMBER OF CHARACTERS TO MOVE
C
C     NOTE THAT FROM AND TO ARE EITHER 1 OR 4-CHARACTER WORDS
C
      DIMENSION  CHARF(4),  CHART(4)
      DATA CHARF, CHART /8*1H   /
C     TEMPORARY LOGICAL UNIT DEFINED
      LUN = 4
C     READ CONTENTS OF SOURCE WORD FROM TO TEMPORARY ARRAY
      WRITE(LUN,1)  FROM
    1 FORMAT(A4)
      BACKSPACE LUN
      READ(LUN,2) CHARF
    2 FORMAT(4A1)
      BACKSPACE LUN
      WRITE(LUN,1)  TO
      BACKSPACE LUN
      READ(LUN,2) (CHART(I),I=1,4)
C     TRANSFER CONTENTS TO TEMPORARY ARRAY
      DO 10  I = 1,NCHR
      CHART(ITC+I-1) = CHARF(IFC+I-1)
   10 CONTINUE
      BACKSPACE LUN
      WRITE(LUN,2) (CHART(I),I=1,4)
      BACKSPACE LUN
      READ(LUN,1)  TO
      RETURN
      END


      SUBROUTINE MDATE(I)
      DIMENSION I(1)
      DATA IBLK /4H    /
      I(1) = IBLK
      I(2) = IBLK
      RETURN
      END


      SUBROUTINE NEWPAG (TITLE,LSKIP,IDATE)
      DIMENSION      TITLE(20),            IDATE(1)
      DATA NPAGE,  LOUT/0,  6/
      WRITE(LOUT,1)
      IF (LSKIP.EQ.0) GO TO 110
      DO 100 I=1,LSKIP
      WRITE(LOUT,2)
  100 CONTINUE
  110 NPAGE=NPAGE+1
      CALL SECOND (A)
      WRITE(LOUT,3)  TITLE,  A,  IDATE(1),  NPAGE
      RETURN
C
    1 FORMAT (1H1)
    2 FORMAT (1H )
    3 FORMAT (1H ,1X,20A4,1X,9HCP TIME =,F8.2,2X,5HDATE ,2X,A10,2X,5HPAG
     1E    ,I4)
      END


      SUBROUTINE PREDAT
C
C        *PREDAT*  TRANSFERS DATA-CARD-IMAGES FROM INPUT(TAPE5)
C        TO TAPE3 AND OUTPUT(TAPE6).
C        PREDAT RETURNS WHEN IT READS THE WORD *MORE* IN COL. 1-4
C        OR WHEN IT FINDS AN EOF.
C
      DIMENSION IMAGE(20)
      DATA LOT,LOUT,ARO/3,6,1H*/
      DATA LIN /5/
      DATA  MORE/4HMORE/
      REWIND LOT
      LYN = 0
100   CONTINUE
C    ******* REMOVE THE NEXT TWO CARDS FOR UNIVAC/IBM *******
      READ (LIN,1) IMAGE
      IF (EOF(LIN)) 120,110
C     ****** INSERT THE NEXT CARD INSTEAD *****
C
C     READ(LIN,1,END=120) IMAGE
110   CONTINUE
      IF (MOD(LYN,40).EQ.0) WRITE (LOUT,2)
      IF (MOD(LYN,10).EQ.0) WRITE (LOUT,3) (K,K=1,80,10),(ARO,K=1,8)
      LYN = LYN+1
      WRITE (LOUT,4) LYN,IMAGE
      WRITE (LDT,1) IMAGE
      IF (IMAGE(1).NE.MORE) GO TO 100
120   CONTINUE
      REWIND LDT
      WRITE (LOUT,5)
      RETURN
1     FORMAT (20A4)
2     FORMAT (1H1,8X,20HIMAGES OF DATA-CARDS)
3     FORMAT (1H010X8I10/14X6HCARD  A1,8(9H.........,A1))
4     FORMAT (I18,2X,20A4)
5     FORMAT (1H1)
      END


      SUBROUTINE SECOND(A)
      A = 0.0
      RETURN
      END


      SUBROUTINE SETPLT(A,B,C,D)
C
C     GENERAL-PURPOSE PRINTER-PLOT ROUTINE
C
      COMMON /IMAG/ IMAGE
      DIMENSION  IMAGE(26,51),                      LABX(5,6),LABEL(5)
      EQUIVALENCE  (IC,CI)
      LOGICAL CONFRM
      DATA          MAXX,MAXY,ARO/104,51,1H+/
      DATA          IBLNK,IDASH     /4H     ,4H----/
C
C     NCHAR IS THE NUMBER OF CHARACTERS PER WORD
      DATA          NCHAR/4/
C
C
C
C-----FOR THIS ENTRY, CALL SETPLOT (X-LOW,
C                                   X-HIGH,
C                                   Y-LOW,
C                                   Y-HIGH)
C         THIS IS THE INITIALIZING CALL - THE ARGUMENTS DEFINE THE
C              ENDS OF THE AXES TO BE PLOTTED
C
      ENTRY SETPLUT
      CONFRM = .FALSE.
 100  DO 110 J = 1,51
      DO 110 I = 1,26
      IMAGE(I,J) = IBLNK
 110  CONTINUE
      DO 120 I = 1,26
 120  IMAGE(I,1) = IDASH
      DO 130 J = 1,MAXY
      CALL MCHAR(1,1HI,1,IMAGE(1,J),1)
 130  CONTINUE
      DX = (B-A)/100.
      DY = (D-C)/(MAXY-1)
      XORG = A
      YORG = C
      IF (.NOT.CONFRM) GO TO 140
      IF (DX.LT..6*DY) DX = .6*DY
      IF (DY.LT.5./3.*DX) DY = 5./3.*DX
 140  KVAL = IBLNK
      KGO = 19
      DO 160 J = 1,6
      VAL = XORG+DX*20*(J-1)
      GO TO 230
 150  LABX(1,J) = LABEL(1)
      LABX(2,J) = LABEL(2)
      LABX(3,J) = LABEL(3)
      LABX(4,J) = LABEL(4)
 160  LABX(5,J) = LABEL(5)
      RETURN
C
C
C-----THIS ENTRY IS IDENTICAL TO SETPLOT, EXCEPT IT IS USED FOR
C                   PLOTTING X AND Y TO SAME SCALE
C                   (A COMPUTED CIRCLE IS PLOTTED AS A CIRCLE)
C
C     ENTRY ISOPLT(A,B,C,D)   FOR UNIVAC OR IBM
      ENTRY ISOPLT
      CONFRM = .TRUE.
      GO TO 100
C
C
C-----FOR THIS ENTRY, CALL GETDXDY (DX,
C                                   DY)
C        USE THIS ENTRY IF YOU NEED TO KNOW THE X- AND Y-INCREMENTS
C             WHICH HAVE BEEN COMPUTED BY ISOPLOT
C
C     ENTRY GTDXDY (A,B)   FOR THE UNIVAC OR IBM
      ENTRY GTDXDY
      A = DX
      B = DY
      RETURN
C
C
C-----FOR THIS ENTRY, CALL PLOTPNT (X-COORD,
C                                   Y-COORD,
C                                   CHARACTER)
```

```
C           CHARACTER MUST BE THE FIRST ONE IN THE WORD, E.G., 1HA.
C          THIS IS THE ENTRY WHICH ENTERS A POINT IN THE PLOT
C
C     ENTRY PLTPNT(A,B,C)   FOR THE UNIVAC OR IBM
      ENTRY PLTPNT
      MGO = 1
      CI = C
 170  JX = (A-XORG)/DX+1.5
      JY = (B-YORG)/DY+1.5
      IF (JX.GT.MAXX.OR.JX.LE.0.OR.JY.GT.MAXY.OR.JY.LE.0) RETURN
      IR = JX/NCHAR+1
      JZ = MOD(JX,NCHAR)
      IF (JZ.GT.0) GO TO 180
      IR = IR-1
      JZ = NCHAR
 180  CONTINUE
      IF (MGO.EQ.2) GO TO 190
      CALL MCHAR(1,CI,JZ,IMAGE(IR,JY),1)
      RETURN
C
C
C-----FOR THIS ENTRY, CALL GETPNT (X-COORD,
C                                  Y-COORD,
C                                   CHARACTER)
C          CHARACTER IS RETURNED IN THE FIRST POSITION OF VARIABLE -C-.
C                 USE THIS ENTRY TO FIND OUT THE CHARACTER
C                 NOW AT (X,Y)
C
C     ENTRY GETPNT(A,B,C)   FOR THE UNIVAC OR IBM
      ENTRY GETPNT
      MGO = 2
      C = 0.
      GO TO 170
 190  CALL MCHAR(JZ,IMAGE(IR,JY),1,C,1)
      RETURN
C
C
C-----FOR THIS ENTRY, CALL PLOTXP1 (CHARACTER)
C          CHARACTER MUST BE THE FIRST ONE IN THE WORD, E.G., 1HA.
C         USE THIS ENTRY TO ADD A CHARACTER TO THE RIGHT OF
C             THE LAST MARK ENTERED
C
C     ENTRY [illegible]PX1(A)   FOR THE UNIVAC OR IBM
      ENTRY PLTPX1
      JX = JX+1
      IF (JX.GT.MAXX) RETURN
      CI = A
      IR = JX/NCHAR+1
      JZ = MOD(JX,NCHAR)
      IF (JZ.GT.0) GO TO 200
      IR = IR-1
      JZ = NCHAR
 200  CONTINUE
      CALL MCHAR(1,CI,JZ,IMAGE(IR,JY),1)
      RETURN
```

```
C
C
C-----FOR THIS ENTRY, CALL PLOTYM1 (CHARACTER)
C          CHARACTER MUST BE THE FIRST ONE IN THE WORD, E.G., 1HA.
C         USE THIS ENTRY TO ADD A CHARACTER BELOW
C              THE LAST MARK ENTERED
C
C     ENTRY PLTYM1(A)   FOR THE UNIVAC OR IBM
      ENTRY PLTYM1
      JY = JY-1
      IF (JY.LE.0) RETURN
      CI = A
      CALL MCHAR(1,CI,JZ,IMAGE(IR,JY),1)
      RETURN
C
C
C         YOU MUST CALL PLOTOUT TO GET PLOT PRINTED
C-----FOR THIS ENTRY, CALL PLOTOUT (PRINT-TAPE)
C
C     ENTRY PLTOUT(A)     FOR THE UNIVAC OR IBM
      ENTRY PLTOUT
      CI = A
      IF (IC.LE.0.OR.IC.GT.49) IC = 6
      WRITE (IC,1)
      KVAL = IARD
      KGO = 14
      DO 220 JYY = 1,MAXY
      JY = MAXY+1-JYY
      VAL = YORG+DY*(JY-1)
      LABEL(1) = IBLNK
      LABEL(2) = IBLNK
      LABEL(3) = IBLNK
      LABEL(4) = IBLNK
      LABEL(5) = IBLNK
      IF (MOD(JYY,10).EQ.1) GO TO 230
210   WRITE (IC,2) LABEL,(IMAGE(I,JY),I=1,26)
220   CONTINUE
      WRITE (IC,3) (ANO,I=1,6),LABX
      RETURN
C
230   V = ABS(VAL)
      IF (V.LT.1.E-76) GO TO 240
      IF (V.GT.9.E4.OR.V.LT.1.E-2) GO TO 250
240   WRITE (4,4) VAL,KVAL
      GO TO 260
250   WRITE (4,5) VAL,KVAL
260   CONTINUE
      BACKSPACE 4
      READ (4,6) LABEL
      IF (KGO.EQ.19) GO TO 150
      IF (KGO.EQ.14) GO TO 210
      STOP
C
1     FORMAT (1H1)
2     FORMAT (31A4)
```

```
3     FORMAT (20X,A1, 5(19X,A1)/8X,30A4/)
4     FORMAT (F17.4,2X,A1)
5     FORMAT (E17.4,2X,A1)
6     FORMAT (5A4)
      END



      SUBROUTINE SOLAR (SLA,SLO,TZ,IY,IM,ID,TIME,D,NV)
C***
C***     SLA...  LATITUDE (DEG)  SOUTH = MINUS
C***     SLO...  LONGITUDE (DEG)  EAST = MINUS
C***     TZ...  TIME ZONE
C***            ALSO INCLUDES FRACTION IF LOCAL TIME IS NOT
C***             STANDARD MERIDIAN TIME.  E.G. POONA, INDIA 5.5
C***     IY..  YEAR
C***     IM..  MONTH
C***     ID..  DAY
C***     TIME.. LOCAL STANDARD TIME IN HOURS AND MINUTES.
C***             1 30 PM = 1330  ** STANDARD TIME **
C***     D..  RETURNED VALUE
C***     NV..  VALUE TO BE RETURNED, SELECTED AS FOLLOWS....
C***           1...  DECLINATION (DEG.)
C***           2...  EQUATION OF TIME ADJUSTMENT (HRS.)
C***           3...  TRUE SOLAR TIME (HRS.)
C***           4...  HOUR ANGLE (DEG.)
C***           5...  SOLAR ELEVATION (DEG.)
C***           6...  OPTICAL AIRMASS
C***     0 ) NV ) 7.  OTHERWISE, D = 9999.
C***
      DIMENSION MD(11)
      DATA MD/31,29,31,30,31,30,2*31,30,31,30/
      DATA A,B,C,SIGA/0.15,3.885,1.253,279.9348/
      RAD=572957.75913E-4
      SDEC=39784.988432E-5
      RE=1.
      IF(SLO.LT.0.) RE=-1.
      KZ=TZ
      TC=(TZ-KZ)*RE
      TZZ=KZ*RE
      SLB=SLA/RAD
      K=ID
      TIMH=TIME/100.
      I=TIMH
      TIMLOC=(TIMH-I)/0.6+I+TC
      IMC=IM-1
      IF(IMC.LT.1) GOTO2
      DO1I=1,IMC
    1 K=K+MD(I)
    2 LEAP=1
      NL=MOD(IY,4)
      IF(NL.LT.1) LEAP=2
      SMER=TZZ*15.
      TK=((SMER-SLO)*4.)/60.
      KR=1
      IF(K.GE.61.AND.LEAP.LT.2) KR=2
      DAD=(TIMLOC+TZZ)/24.                                              UL
      DAD=DAD+K-KR                                                      UL
      DF=DAD*360./365.242                                               UL
      DE=DF/RAD                                                         UL
      DESIN=SIN(DE)                                                     UL
      DECOS=COS(DE)                                                     UL
      DESIN2=SIN(DE*2.)                                                 UL
      DECOS2=COS(DE*2.)                                                 UL
      SIG=SIGA+DF+1.914827*DESIN-0.079525*DECOS+0.019938*DESIN2-0.00162*UL
     SDECOS2                                                            UL
      SIG=SIG/RAD                                                       UL
      DECSIN=SDEC*SIN(SIG)                                              UL
      EFFDEC= ASIN(DECSIN)                                              UL
      IF(NV.NE.1) GOTO10                                                UL
      D=EFFDEC*RAD                                                      UL
      RETURN                                                            UL
   10 EQT=0.12357*DESIN-0.004289*DECOS+0.153809*DESIN2+0.060783*DECOS2  UL
      IF(NV.NE.2) GOTO11                                                UL
      D=EQT                                                             UL
      RETURN                                                            UL
   11 TST=TK+TIMLOC-EQT                                                 UL
      IF(NV.NE.3) GOTO12                                                UL
      D=TST                                                             UL
      IF(D.LT.0.) D=D+24.                                               UL
      IF(D.GE.24.) D=D-24.                                              UL
      RETURN                                                            UL
   12 HRANGL=ABS(TST-12.)*15.                                           UL
      IF(NV.NE.4) GOTO13                                                UL
      D=HRANGL                                                          UL
      RETURN                                                            UL
   13 HRANGL=HRANGL/RAD                                                 UL
      SOLSIN=DECSIN*SIN(SLB)+COS(EFFDEC)*COS(SLB)*COS(HRANGL)           UL
      SOLEL= ASIN(SOLSIN)*RAD                                           UL
      IF(NV.NE.5) GOTO14                                                UL
      D=SOLEL                                                           UL
      RETURN                                                            UL
   14 IF(NV.NE.6) GOTO8                                                 UL
      IF(SOLEL.LE.0.) GOTO8                                             UL
      TK=SOLEL+B                                                        UL
      E=1./TK**C                                                        UL
      D=1./(A*E+SOLSIN)                                                 UL
      RETURN                                                            UL
    8 D=9999.                                                           UL
      RETURN                                                            UL
      END                                                               UL
                                                                        UL

      SUBROUTINE XMIT(N,A,B)
C                                                                       UL
C         IF  N  POSITIVE, TRANSMITS  N  WORDS FROM  A  TO  B           UL
C         IF  N  NEGATIVE, TRANSMITS  A  TO  N  WORDS OF  B             UL
C                                                                       UL
      DIMENSION A(2),B(2)                                               UL
C                                                                       UL
      IF (N) 100,120,120                                                UL
100   K = IABS(N)                                                       UL
      DO 110 I = 1,K                                                    UL
      B(I) = A(1)                                                       UL
110   CONTINUE                                                          UL
      RETURN                                                            UL
C                                                                       UL
120   DO 130 I = 1,N                                                    UL
      B(I) = A(I)                                                       UL
130   CONTINUE                                                          UL
      RETURN                                                            UL
C                                                                       UL
      END                                                               UL
                                                                        UL
```

APPENDIX D

AN IMPROVED PHOTOCHEMICAL MECHANISM AND RELATED MODIFICATIONS FOR THE CHEMICAL-DIFFUSION MODULE

1. Description of Chemical Mechanism and Related Modifications

Concurrent with this study, an improved chemical mechanism was developed by ERT for use in the trajectory model. This appendix describes the new mechanism and related changes in the chemical-diffusion module for its use. Changes in the FORTRAN source code and data sets for the KEMOD program are included.

The chemical reactions and chemical species included in the improved mechanism are listed in Tables D-1 and D-2. The major differences between it and the mechanism described in previous sections are as follows:

1) A distinction is made between ethylene and higher olefinic hydrocarbon compounds to account for their differences in reactivity.

2) The ozone-olefin mechanism reflects the formation of more stable products and fewer radical species.

3) The aromatic hydrocarbon photooxidation mechanism is more reliable.

The development of this mechanism and the results for its evaluation relative to smog chamber data are described in Lloyd et al. 1979.

In adapting this mechanism to the atmosphere, new kinetic data and new assumptions have been employed in determining certain chemical reaction rate constants. The rate constants for photolysis of nitrogen dioxide, nitrous acid, formaldehyde, and higher aldehydes by solar ultraviolet radiation have been updated using new solar actinic flux, quantum yield, and absorption cross-section data (Dermerjian et al. 1979, Moortgat et al. 1978). The new clear-sky photolysis rates are shown in Tables D-3 through D-6 as a function of solar zenith angle and elevation.

TABLE D-1

THE PHOTOCHEMICAL REACTION MECHANISM

	Reactions	Rate Constant
1	NO2 + HV = 0 + NO	Radiation Dependent
2	0 + 02 + M = 03 + M	4.12E+06
3	03 + NO = NO2 + 02	2.50E+01
4	NO + NO2 + N20 = 2HONO	2.20E-09
5	2HONO = NO + NO2 + H20	1.40E-03
6	HONO + HV = OH + NO	Radiation Dependent
7	OH + NO + M = HONO + M	1.44E+04
8	OH + NO2 + M = HNO3 + M	1.44E+04
9	OH + CO = HO2 + CO2	4.40E+02
10	HO2 + NO = NO2 + OH	1.20E+04
11	HO2 + NO2 = HNO4	1.71E+03
12	2HO2 = H2O2 + 02	3.61E+03
13	NO2 + 03 = NO3 + 02	5.00E-02
14	NO3 + NO = 2NO2	2.70E+04
15	NO3 + NO2 = N2O5	9.30E+02
16	N2O5 + H20 = 2HNO3	1.00E-06
17	N2O5 = NO3 + NO2	Temperature Dependent
18	OH + ALKE = AO2	Elevation Dependent
19	OH + C2H4 = AO2	Elevation Dependent
20	AO2 + NO = NO2 + AO	2.90E+04
21	AO + 02 = .5RCHO + 1.5HCHO + HO2	4.10E+05
22	HNO4 = HO2 + NO2	Temperature Dependent
23	0 + ALKE = .3EPOX + .3RCHO + .4HO2 + .4RO2	Elevation Dependent
24	0 + C2H4 = .3EPOX + .3RCHO + .4HO2 + .4RO2 + .4CO	Elevation Dependent
25	03 + ALKE = .5HCHO + .4RCHO + .4HD + .4RD + .1RO2	Elevation Dependent
26	03 + C2H4 = HCHO + .8HD + .4HO2 + .2CO2	Elevation Dependent
27	HD + NO = HCHO + NO2	2.90E+04

TABLE D-1 (Continued)

	Reactions	Rate Constant
28	RD + NO = RCHO + NO2	2.90E+04
29	HD + NO2 = HCHO + NO3	1.90E+04
30	RD + NO2 = RCHO + NO3	1.90E+04
31	HD + HCHO = OZID	1.00E+01
32	HD + RCHO = OZID	1.00E+01
33	RD + HCHO = OZID	1.00E+01
34	RD + RCHO = OZID	1.00E+01
35	OH + PA = H2O + PAO2	3.80E+03
36	PAO2 + NO = NO2 + .85PAO + .15RO2	2.90E+04
37	PAO2 + NO = NTRA	2.60E+03
38	PAO = RO2 + .5HCHO + .5RCHO	1.40E+05
39	RO2 + NO = NO2 + PAO	2.90E+04
40	PAO + O2 = RCHO + HO2	6.70E+04
41	PAO + NO2 = .85NTRA + .15RCHO + .15HONO	2.30E+03
42	OH + RCHO = RCO3 + H2O	2.20E+04
43	RCHO + HV = RO2 + HO2 + CO	Radiation Dependent
44	RCO3 + NO = CO2 + NO2 + RO2	2.90E+04
45	RCO3 + NO2 = PAN	1.70E+04
46	PAN = RCO3 + NO2	Temperature Dependent
47	HCHO + HV = HO2 + CO	Radiation Dependent
48	HCHO + OH = H2O + HO2 + CO	1.60E+04
49	RO2 + HO2 = RO2H + O2	4.20E+03
50	OH + AR = HO2 + AC	1.67E+04
51	OH + AR = ARO	3.33E+03
52	OH + AC = ARP + HO2	4.90E+04
53	ARO + NO = NO2 + ARIN	2.90E+04
54	ARO + NO2 = ANO2	1.90E+04
55	ARO + HO2 = AC + H2O2	1.80E+03
56	ARIN = RO2 + OH	1.00E-01
57	AR + OH = H2O + ABO	5.00E+03
58	ABO + NO = NO2 + HO2 + ACHO	2.90E+04

TABLE D-1 (Continued)

	Reactions	Rate Constant
59	$SO_2 + OH \rightarrow SO_4$	1.76E+03
60	$SO_2 + HO_2 \rightarrow SO_4\ OH$	3.00E-02
61	$SO_2 + HD \rightarrow SO_4 + HCHO$	2.90E+03
62	$SO_2 + RD \rightarrow SO_4 + RCHO$	2.90E+03
63	$SO_2 + RO_2 \rightarrow SO_4 + PAO$	8.00E+00
64	$SO_2 + AO_2 \rightarrow SO_4 + AO$	8.00E+00

TABLE D-2

CHEMICAL SPECIES SYMBOL DEFINITIONS

Species	Symbol Designation	Species No.
AO	Alkoxy radical equivalent of AO_2	17
AO_2	Product of OH addition to olefin in the presence of O_2	18
AR	Aromatic hydrocarbons	12
AC	Product of OH addition to aromatic hydrocarbon followed by H atom abstraction by O_2.	14
ARO	Product of addition of OH to a cresol in the presence of O_2	13
ARP	Product of OH addition to AC followed by H atom abstraction by O_2.	-
ABO	Product of H-abstraction from side chain alkyl group of benzene ring followed by addition of O_2 to radical formed	16
ARIN	Intermediate formed from reaction of ARO with NO, forming NO_2	15
ACHO	Aromatic aldehyde	39
ANO_2	Aromatic nitro compound	38
AROH	Cresol	-
CO	Carbon monoxide	5
CO_2	Carbon dioxide	-
HD	Criegee intermediate ($HCHO_2$)	25
RD	Criegee intermediate ($RCHO_2$)	26
EPOX	Epoxide formed from O atom addition to olefin	-
HCHO	Formaldehyde	6
HONO	Nitrous acid	4
HNO_3	Nitric acid	34

TABLE D-2 (Continued)

Species	Symbol Designation	Species No.
HNO_4	Pernitric acid, HO_2HO_2	20
HO_2	Hydroperoxyl radical	22
H_2O	Water	33
H_2O_2	Hydrogen peroxide	36
HV	Photon	-
M	Any third body, such as N_2 or O_2	-
NO	Nitric oxide	1
NO_2	Nitrogen dioxide	2
NO_3	Nitrate radical	19
N_2O_5	Dinitrogen pentoxide	21
NTRA	Organic nitrate	37
O	Oxygen atom (ground state)	32
O_2	Oxygen	-
O_3	Ozone	3
OH	Hydroxyl radical	31
OZID	Ozonides	35
C_2H_4	Ethene	9
ALKE	Alkenes other than ethene	8
PA	Alkanes (paraffinic hydrocarbons)	10
PAN	Peroxyacetyl Nitrate	11
PAO	Alkoxy radical formed by PA	27
PAO_2	Alkyl peroxy radical from the O_2 addition to the radical formed by H-abstraction from a paraffinic hydrocarbon	28
R	Generalized alkyl group (e.g., C_2H_5, C_3H_7, etc.)	-
RCHO	Aldehydes other than formaldehyde	7
RCO_3	Acyl peroxy radical	24

TABLE D-2 (Continued)

Species	Symbol Designation	Species No.
RO	Alkoxyl radical	-
RO_2	Alkyl peroxy radical	23
RO_2H	Product of disproportionation between HO_2 and RO_2	-
SO_2	Sulfur dioxide	29
SO_4	Sulfate	30

TABLE D-3

ATMOSPHERIC PHOTODISSOCIATION RATES FOR RCHO (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	1.380E-03	1.494E-03	1.602E-03	1.716E-03	1.836E-03	1.944E-03	2.046E-03	2.382E-03
10.0	1.362E-03	1.476E-03	1.584E-03	1.698E-03	1.806E-03	1.914E-03	2.202E-03	2.358E-03
20.0	1.296E-03	1.404E-03	1.512E-03	1.620E-03	1.734E-03	1.842E-03	1.938E-03	2.274E-03
30.0	1.176E-03	1.284E-03	1.386E-03	1.494E-03	1.602E-03	1.704E-03	1.800E-03	2.130E-03
40.0	1.014E-03	1.116E-03	1.212E-03	1.308E-03	1.410E-03	1.506E-03	1.596E-03	1.920E-03
50.0	8.040E-04	8.820E-04	9.720E-04	1.062E-03	1.146E-03	1.236E-03	1.314E-03	1.614E-03
60.0	5.520E-04	6.180E-04	6.840E-04	7.500E-04	8.160E-04	8.820E-04	9.540E-04	1.212E-04
70.0	2.820E-04	3.240E-04	3.600E-04	4.020E-04	4.440E-04	4.860E-04	5.280E-04	7.080E-04
78.0	1.140E-04	1.260E-04	1.380E-04	1.560E-04	1.740E-04	1.920E-04	2.100E-04	3.000E-04
86.0	1.800E-05	2.400E-05	2.400E-05	2.400E-05	3.000E-05	3.000E-05	3.000E-05	3.600E-05

TABLE D-4

ATMOSPHERIC PHOTODISSOCIATION RATES FOR NO2 (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	5.340E-01	5.650E-01	5.940E-01	6.210E-01	6.480E-01	6.710E-01	6.910E-01	7.580E-01
10.0	5.290E-01	5.610E-01	5.890E-01	6.170E-01	6.440E-01	6.680E-01	6.880E-01	7.550E-01
20.0	5.160E-01	5.490E-01	5.790E-01	6.070E-01	6.330E-01	6.570E-01	6.780E-01	7.460E-01
30.0	4.930E-01	5.260E-01	5.570E-01	5.860E-01	6.140E-01	6.380E-01	6.500E-01	7.200E-01
40.0	4.570E-01	4.920E-01	5.230E-01	5.530E-01	5.820E-01	6.070E-01	6.290E-01	7.030E-01
50.0	4.040E-01	4.390E-01	4.720E-01	5.030E-01	5.310E-01	5.560E-01	5.790E-01	6.570E-01
60.0	3.250E-01	3.600E-01	3.920E-01	4.230E-01	4.510E-01	4.760E-01	4.990E-01	5.810E-01
70.0	2.140E-01	2.430E-01	2.720E-01	3.000E-01	3.260E-01	3.480E-01	3.690E-01	4.510E-01
78.0	1.050E-01	1.230E-01	1.410E-01	1.600E-01	1.790E-01	1.960E-01	2.110E-01	2.800E-01
86.0	2.300E-02	2.500E-02	2.800E-02	3.100E-02	3.400E-02	3.700E-02	4.000E-02	5.000E-02

TABLE D-5

ATMOSPHERICPHOTODISSOCIATION RATES FOR HCHO* (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	6.140E-03	6.620E-03	7.090E-03	7.560E-03	8.020E-03	8.460E-03	8.860E-03	1.020E-02
10.0	6.060E-03	6.540E-03	7.000E-03	7.470E-03	7.940E-03	8.380E-03	8.780E-03	1.010E-02
20.0	5.810E-03	6.290E-03	6.750E-03	7.210E-03	7.670E-03	8.100E-03	8.500E-03	9.850E-03
30.0	5.380E-03	5.840E-03	6.290E-03	6.750E-03	7.190E-03	7.620E-03	8.010E-03	9.370E-03
40.0	4.740E-03	5.190E-03	5.620E-03	6.050E-03	6.480E-03	6.890E-03	7.260E-03	8.610E-03
50.0	3.880E-03	4.280E-03	4.690E-03	5.080E-03	5.470E-03	5.850E-03	6.200E-03	7.500E-03
60.0	2.800E-03	3.130E-03	3.460E-03	3.790E-03	4.120E-03	4.440E-03	4.740E-03	5.940E-03
70.0	1.560E-03	1.780E-03	1.990E-03	2.220E-03	2.440E-03	2.660E-03	2.880E-03	3.810E-03
78.0	6.700E-04	7.600E-04	8.600E-04	9.600E-04	1.070E-03	1.170E-03	1.280E-03	1.810E-03
86.0	1.300E-04	1.500E-04	1.600E-04	1.800E-04	1.900E-04	2.100E-04	2.200E-04	2.900E-04

---

* Radical and molecular product formation paths combined into on reaction:

$$HCHO + h\nu \overset{k_{47}}{=} HO2 + CO$$

TABLE D-6

ATMOSPHERIC PHOTODISSOCIATION RATES FOR HONO (PER MINUTE)

ELEVATIONS IN METERS

Zenith Angle	0.0	150.0	360.0	640.0	980.0	1380.0	1840.0	4210.0
0.0	9.580E-02	1.020E-01	1.080E-01	1.130E-01	1.180E-01	1.230E-01	1.270E-01	1.400E-01
10.0	9.500E-02	1.010E-01	1.070E-01	1.120E-01	1.170E-01	1.220E-01	1.260E-01	1.400E-01
20.0	9.250E-02	9.870E-02	1.050E-01	1.100E-01	1.150E-01	1.200E-01	1.240E-01	1.380E-01
30.0	8.810E-02	9.450E-02	1.000E-01	1.060E-01	1.120E-01	1.160E-01	1.210E-01	1.350E-01
40.0	8.120E-02	8.770E-02	9.380E-02	9.960E-02	1.050E-01	1.100E-01	1.450E-01	1.290E-01
50.0	7.100E-02	7.770E-02	8.370E-02	8.950E-02	9.510E-02	1.000E-01	1.040E-01	1.200E-01
60.0	5.630E-02	6.260E-02	6.850E-02	7.410E-02	7.950E-02	8.430E-02	8.860E-02	1.050E-02
70.0	3.610E-02	4.110E-02	4.600E-02	5.080E-02	5.540E-02	5.960E-02	6.340E-02	7.940E-02
78.0	1.740E-02	2.000E-02	2.290E-02	2.600E-02	2.890E-02	3.170E-02	3.440E-02	4.680E-02
86.0	3.900E-03	4.300E-03	4.800E-03	5.200E-03	5.700E-03	6.100E-03	6.560E-02	8.780E-02

New assumptions concerning the typical reactivity of individual hydrocarbon groups in the atmosphere are employed. The rate constants for reactions involving alkanes (PA) assume that the atmospheric alkane mixture is as reactive as pure n-butane. The atmospheric mixture of aromatic hydrocarbons (AR) is assumed to be as reactive as a 50 percent o-xylene/50 percent m-xylene mixture. For the reactions involving higher olefins (alkenes excluding ethylene) the reactivity of the mixture is assumed to decrease with elevation. It is believed that the most reactive olefins (double bonded C_4's, C_5's, and C_6's), which are emitted near the surface, react so quickly that few are in fact available for diffusion to the higher elevations. In other words, their reaction time scale is significantly shorter than their diffusion time scale. The elevation dependent rate constants for reactions involving higher olefins (ALKE) are shown in Table D-7. In the lowest two cells of the air parcel (typically 0 to 150 meters), the higher olefin mixture is assumed to be as reactive as a 50 percent propene/50 percent cis-2-butene mixture. In the third and fourth vertical cells, the mixture is assumed to react as if it were composed of 75 percent propene/25 percent cis-2-butene. Above the fourth cell, the rate constants for pure propene are assumed.

In this mechanism, the temperature dependence of the dinitrogen pentoxide (N2O5) dissociation rate constant is included. The rate constant for this reaction step is determined from the expression

$$R_{18} = 3.42 \times 10^{16} \exp(-10{,}600/T) \ (\text{min}^{-1})$$

where T is temperature in degrees Kelvin (Baulch et al. 1973).

In order to utilize the two olefin class mechanism with the RAPS emission inventory, the RAPS single olefin emission rates are uniformly partitioned into ethylene and higher olefin emission rates. The single olefin emission rates are partitioned assuming a 50/50 split on a molar basis. This assumes that olefins emissions are approximately 35 percent ethylene and 65 percent higher olefins by weight which is typical of vehicle olefin emissions.

Two additional features have been implemented in the chemical diffusion module for computing certain species concentrations. The concentrations of oxygen atoms (O) and hydroxyl radicals (OH) are computed using algebraic expressions based on the steady-state or chemical equilibrium assumption.

TABLE D-7

HIGHER OLEFIN REACTION RATE CONSTANTS

	Rate Constants		
Reaction	Cells 1&2	Cells 3&4	Cell 5
O3 + ALKE $\xrightarrow{k_{25}}$ .5HCHO + .5RCHO + .4HD + 0.4RD + .1RO2	4.58E-2	2.37E-2	1.60E-3
OH + ALKE $\xrightarrow{k_{18}}$ AO2	5.80E+4	4.75E+4	3.70E+4
O + ALKE $\xrightarrow{k_{23}}$ .3EPOX + .3RCHO + .4HO2 + .4RO2	1.46E+4	9.90E+3	5.20E+3

Both species react so quickly that the assumption is appropriate. By computing their concentrations via explicit algebraic equations instead of numerical integration, the numerical stiffness of the system of differential equations is reduced. This results in computer time savings. Second, the concentrations of six "product only" species (not reactants) are computed in an approximate manner. These species include nitric acid, organic nitrates, aromatic nitrates, aromatic aldehydes, hydrogen peroxide, and ozonides. These predicted concentrations are probably accurate within an order of magnitude and may be of interest to researchers.

Lastly, the manner in which the phenomenon of surface deposition is simulated in the model has been reformulated. The deposition term in the model's equation has been modified so that it is calculated assuming only the concentrations in a ten meter thick surface layer are effected by the phenomenon. Formerly, the effects of surface deposition may have been exaggerated by assuming the concentrations in the lowest 30 to 100 meters of the air parcel were subject to deposition.

2. KEMOD Input Data Modifications

Numerous changes in the KEMOD input deck are required for simulations with the improved chemical mechanism. A listing of the revised portion of the input deck for the sample problem is shown in Figure D-1. This figure includes all the data cards positioned after the emission rate cards, even though only some of the cards differ from that described in Section 5. The general sequence in which the cards are read is, of course, identical to that described in Section 5.

The changes in the data deck are summarized as follows:

1) The number of reaction equals 64.
2) The number of species equals 39.
3) The number of species not formally integrated is 9.
4) There is a 39 card species list.
5) There are some different emission species indices.
6) NO2 is specified as a species with surface deposition.
7) There are different indices for SO2 and SO4 on the deposition cards.
8) There are 39 cards for initial concentration (see Table D-8 for delineation of user specified and internally computed species initial concentrations).
9) There are 64 chemical rate constant cards.
10) There are 64 chemical reaction cards.
11) There are 4 time-varying photolyic reactions and 4 cards indicating their reaction numbers.

TABLE D-8

USER SPECIFIED AND INTERNALLY COMPUTED INITIAL CONCENTRATIONS

User Specified Species	Internally Computed Species
NO	NO2
O3	HONO
CO	PAN
HCHO	ARO
RCHO	AC
ALKE	ARIN
C2H4	ABO
PA	AO
AR	AO2
HO2	NO3
SO2	HNO4
SO4	N2O5
OH	RO2
H2O	RCO3
HNO3*	HD
OZID*	RD
H2O2*	PAO
NTRA*	PAO2
ANO2*	O
ACHO*	

---

*"Products Only" Species

3. KEMOD FORTRAN Source Code Modifications

In order to accommodate the changes related to use of the improved mechanism, many KEMOD subroutines have been modified. In four subroutines (DRIVE, PEDERV, PSET, and TSTEP) the user must change the array sizes in common block CHEM2 so that it appears as follows:

```
COMMON/CHEM2/  CONIN (40, 5), WTMOLE (40), RATKON (65), RATEFF (65),
1              RATEV (4, 5), QRATE, NVRATE, LOCVRT (4)
```

This is the only change in these four routines. A new subroutine named PHOTOR has been added to replace the tasks formerly performed by subroutines PHOTOD and UPRAT2. The names of subroutines with significant changes are shown below. Complete listings of these subroutines appear on the following pages.

DIFFUN
ISTATE
JACOB
KEMOD2
PHOTOR
PRODUK
RATEHI
RATES
STEADY
TEMPR
UNMIXR
UPFLX1
UPSORC

```
IMAGES OF DATA-CARDS

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  321  POINT SOURCES 46         735.35  .0000       .3910-008   .0000       .5444-009
  322                                   .0000       .0000       .0000
  323                46      .5037+000  .9812-001   .1570+000   .2412+000   .0000
  324  POINT SOURCES 47         735.84  .9919-008   .7141-011   .7141-011   .2380-011
  325                                   .7141-011   .1204-009   .1860-010
  326                47      .3773-001  .5420+000   .1515+000   .2687+000   .0000
  327  POINT SOURCES 48         735.84  .7541-008   .0000       .0000       .0000
  328                                   .0000       .0000       .0000
  329                48      .5250+000  .1023+000   .1635+000   .2392+000   .0000
  330  POINT SOURCES 49         736.08  .8620-008   .0000       .0000       .0000

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  331                                   .0000       .0000       .0000
  332                49      .5249+000  .1023+000   .1635+000   .2092+000   .0000
  333  POINT SOURCES 50         736.22  .7189-008   .0000       .0000       .0000
  334                                   .0000       .0000       .3288-011
  335                50      .5265+000  .1026+000   .1641+000   .2069+000   .0000
  336  POINT SOURCES 51         742.74  .1731-008   .1503-011   .1503-011   .5010-012
  337                                   .1503-011   .2086-009   .8867-010
  338                51      .4026-001  .5771+000   .1610+000   .2216+000   .0000
  339  POINT SOURCES 52         742.74  .1731-008   .1503-011   .1503-011   .5010-012
  340                                   .1503-011   .2085-009   .8867-010

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  341                52      .4026-001  .5771+000   .1610+000   .2216+000   .0000
  342  POINT SOURCES 53         742.87  .6204-008   .3006-011   .3006-011   .1002-011
  343                                   .3006-011   .3216-009   .9149-009
  344                53      .4120-001  .5893+000   .1645+000   .2043+000   .0000
  345  POINT SOURCES 54         752.32  .6287-009   .4527-012   .4527-012   .1509-012
  346                                   .4527-012   .7635-010   .1179-011
  347                54      .5265+000  .1026+000   .1641+000   .2069+000   .0000
  348  POINT SOURCES 55         -10.00  .0000       .0000       .0000       .0000
  349                                   .0000       .0000       .0000
  350  INITIAL INTEGRATION TIME STEP SIZE         1.0E-04     (MINUTES)

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  351  MAXIMUM INTEGRATION TIME STEP SIZE      6.0             (MINUTES)
  352  INTEGRATION ERROR CONTROL CRITERIA         1.0E-03
  353  UPDATE INTERVAL                         15.0            (MINUTES)
  354  PRINT INTERVAL                          30.0            (MINUTES)
  355  DO WE OUTPUT INTEGRATION PARAMETERS<    YES
  356  DO WE GENERATE PHOTODISSOCIATION RATES< YES
  357  SHOULD THEY VARY WITH ELLVATION<        YES
  358  DO WE HAVE SKY CLEARNESS RATIOS INPUT<  YES
  359  DO WE HAVE TEMPERATURES INPUT<          YES
  360  SHOULD WE PUNCH GROUND CONCENTRATIONS<  NO
```

Figure D-1. KEMOD Input Deck for Improved Chemical Mechanism

```
IMAGES OF DATA-CARDS

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 361    MAXIMUM NUMBER OF INTEGRATION STEPS                 1000
 362    NUMBER OF CHEMICAL REACTIONS                          64
 363    NUMBER OF SPECIES                                     33
 364    NUMBER OF SPECIES NOT INTEGRATED                       9
 365    NUMBER OF VERTICAL MESH POINTS                         5
 366    ELEVATION OF FIRST  VERTICAL MESH POINT 0.0                (METERS ABOVE SURFACE)
 367    ELEVATION OF SECOND VERTICAL MESH POINT 120.0
 368    ELEVATION OF THIRD  VERTICAL MESH POINT 300.0
 369    ELEVATION OF FOURTH VERTICAL MESH POINT 600.0
 370    ELEVATION OF FIFTH  VERTICAL MESH POINT 1200.0

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 371    SPECIES NAME, MOLE WT, AND BOUND COND   NO        30.
 372                                     2      NO2       46.                 1
 373                                     3      O3        48.                 1
 374                                     4      HONO      47.
 375                                     5      CO        28.
 376                                     6      HCHO      30.
 377                                     7      RCHO      58.
 378                                     8      ALKE      49.
 379                                     9      C2H4      28.
 380                                    10      PA        58.

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 381                                    11      PAN       121.
 382                                    12      AR        92.
 383                                    13      ARO       123.
 384                                    14      AC        100.
 385                                    15      ARIN      100.
 386                                    16      A30       100.
 387                                    17      AO        58.
 388                                    18      AO2       74.
 389                                    19      NO3       62.
 390                                    20      HNO4      79.

        1         11        21        31        41        51        61        71
CARD    +.........+.........+.........+.........+.........+.........+.........+.........
 391                                    21      N2O5      108.
 392                                    22      HO2       33.
 393                                    23      RO2       89.
 394                                    24      RCO3      103.
 395                                    25      HO        74.
 396                                    26      RO        73.
 397                                    27      PAO       73.
 398                                    28      PAO2      89.
 399                                    29      SO2       64.
 400                                    30      SO4       96.
```

Figure D-1 (Continued)

```
IMAGES OF DATA-CARDS

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  401                                 31        OH        17.
  402                                 32        O         16.
  403                                 33        H2O       18.
  404                                 34        HNO3      63.
  405                                 35        OZID      50.
  406                                 36        H2O2      34.
  407                                 37        NTRA      80.
  408                                 38        ANO2      100.
  409                                 39        ACHO      70.
  410  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 1          1

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  411  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 2         10
  412  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 3          9
  413  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 4         12
  414  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 5          5
  415  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 6          5
  415  SPECIE  INDEX FOR AREA SOURCE FLUX NO. 7         29
  417  SPECIE INDEX FOR POINT SOURCE FLUX NO. 1          1
  418  SPECIE INDEX FOR POINT SOURCE FLUX NO. 2         10
  419  SPECIE INDEX FOR POINT SOURCE FLUX NO. 3          9
  420  SPECIE INDEX FOR POINT SOURCE FLUX NO. 4         12

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  421  SPECIE INDEX FOR POINT SOURCE FLUX NO. 5          5
  422  SPECIE INDEX FOR POINT SOURCE FLUX NO. 6          5
  423  SPECIE INDEX FOR POINT SOURCE FLUX NO. 7         29
  424  DEPOSITION SPECIE INDEX,VELOCITY,EXP.             2     0.60       1.0
  425  DEPOSITION SPECIE INDEX,VELOCITY,EXP.             3     0.30       1.0
  426  DEPOSITION SPECIE INDEX,VELOCITY,EXP.            29     0.60       1.0
  427  DEPOSITION SPECIE INDEX,VELOCITY,EXP.            33     0.18       1.0
  428  NO     IC 1            3.0E-03   2.5E-03   1.5E-03   1.5E-03   1.5E-03
  429  NO2     2 COMPUTED
  430  O3     IC 3         .029      .043      .058      .058      .058

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  431  HONO    4 COMPUTED
  432  CO     IC 5         .133      .110      .066      .065      .066
  433  HCHO   IC 6            4.0E-04   3.4E-04   2.0E-04   2.0E-04   2.0E-04
  434  RCHO   IC 7            2.0E-04   1.7E-04   1.0E-04   1.0E-04   1.0E-04
  435  ALKE   IC 8            2.0E-04   1.7E-04   1.0E-04   1.0E-04   1.0E-04
  436  C2H4   IC 9            4.0E-04   3.4E-04   2.0E-04   2.0E-04   2.0E-04
  437  PA     IC 10           3.0E-03  2.54E-03   1.5E-03   1.5E-03   1.5E-03
  438  PAN     11 COMPUTED
  439  AR     IC 12           6.0E-04  5.10E-04   3.0E-04   3.0E-04   3.0E-04
  440  ARO    13 COMPUTED
```

Figure D-1 (Continued)

```
IMAGES OF DATA-CARDS

        1         11          21         31         41         51         61         71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 441   AC        14COMPUTED
 442   APIN      15COMPUTED
 443   ARO       16COMPUTED
 444   AO        17COMPUTED
 445   AO2       18COMPUTED
 446   NO3       19COMPUTED
 447   HNO4      20COMPUTED
 448   N2O5      21COMPUTED
 449   HO2     IC 22              1.0E-06    1.0E-06    1.0E-06    1.0E-06    1.0E-06
 450   RO2       23COMPUTED

        1         11          21         31         41         51         61         71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 451   RCO3      24COMPUTED
 452   HO        25COMPUTED
 453   RO        26COMPUTED
 454   PAO       27COMPUTED
 455   PAO2      28COMPUTED
 456   SO2    IC  29              1.0E-02    .85E-03    .50E-03    .50E-03    .50E-03
 457   SO4     IC 30              1.0E-03    .85E-03    .50E-03    .50E-03    .50E-03
 458   OH      IC 31              1.00E-8    1.00E-8    1.00E-8    1.00E-8    1.00E-8
 459   O         32COMPUTED
 460   H2O     IC 33              1.7E 04    1.7E 04    1.7E 04    1.7E 04    1.7E 04

        1         11          21         31         41         51         61         71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 461   HNO3  IC   34               1.E-09     1.E-09     1.E-09     1.E-09     1.E-09
 462   OZID  IC   35               1.E-09     1.E-09     1.E-09     1.E-09     1.E-09
 463   H2O2  IC   36               1.E-09     1.E-09     1.E-09     1.E-09     1.E-09
 464   NTRA  IC   37               1.E-09     1.E-09     1.E-09     1.E-09     1.E-09
 465   ANO2  IC   38               1.E-09     1.E-09     1.E-09     1.E-09     1.E-09
 466   ACHO  IC   39               1.E-09     1.E-09     1.E-09     1.E-09     1.E-09
 467              REACTION NO.        1                           COMPUTED FROM SOLAR
 468              REACTION NO.        2             4.12E+05
 469              REACTION NO.        3             2.50E+01
 470              REACTION NO.        4             2.20E-03

        1         11          21         31         41         51         61         71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
 471              REACTION NO.        5             1.40E-03
 472              REACTION NO.        6                           COMPUTED FROM SOLAR
 473              REACTION NO.        7             1.44E+04
 474              REACTION NO.        8             1.44E+04
 475              REACTION NO.        9             4.40E+02
 476              REACTION NO.        10            1.20E+04
 477              REACTION NO.        11            1.71E+03
 478              REACTION NO.        12            3.61E+03
 479              REACTION NO.        13            5.00E-02
 480              REACTION NO.        14            2.70E+04
```

Figure D-1 (Continued)

```
IMAGES OF DATA-CARDS

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+........
 481           REACTION NO.           15           9.30E 02
 482           REACTION NO.           15           1.00E-03
 483           REACTION NO.           17                 COMPUTED FROM TEMPERATURE
 484           REACTION NO.           18               COMPUTED FROM MESH PT HEIGHT
 485           REACTION NO.           19           1.20E+04
 486           REACTION NO.           20           2.90E+04
 487           REACTION NO.           21           4.10E+05
 488           REACTION NO.           22                 COMPUTED FROM TEMPERATURE
 489           REACTION NO.           23               COMPUTED FROM MESH PT HEIGHT
 490           REACTION NO.           24           1.10E+03

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+........
 491           REACTION NO.           25               COMPUTED FROM MESH PT HEIGHT
 492           REACTION NO.           26           2.50E-03
 493           REACTION NO.           27           2.90E+04
 494           REACTION NO.           28           2.90E+04
 495           REACTION NO.           29           1.90E+04
 496           REACTION NO.           30           1.90E+04
 497           REACTION NO.           31           1.00E+01
 498           REACTION NO.           32           1.00E+01
 499           REACTION NO.           33           1.00E+01
 500           REACTION NO.           34           1.00E+01

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+........
 501           REACTION NO.           35            4.40E+3
 502           REACTION NO.           36           2.90E+04
 503           REACTION NO.           37           2.60E+03
 504           REACTION NO.           38           1.40E+05
 505           REACTION NO.           39           2.90E+04
 506           REACTION NO.           40           6.70E+04
 507           REACTION NO.           41           2.30E+03
 508           REACTION NO.           42           2.20E+04
 509           REACTION NO.           43                COMPUTED FROM SOLAR
 510           REACTION NO.           44           2.90E+04

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+........
 511           REACTION NO.           45           1.70E+04
 512           REACTION NO.           46                COMPUTED FROM TEMPERATURE
 513           REACTION NO.           47                COMPUTED FROM SOLAR
 514           REACTION NO.           48           1.60E+04
 515           REACTION NO.           49           4.20E+03
 516           REACTION NO.           50            2.24E+4
 517           REACTION NO.           51            5.60E+3
 518           REACTION NO.           52           4.90E+04
 519           REACTION NO.           53           2.90E+04
 520           REACTION NO.           54           1.90E+04
```

Figure D-1 (Continued)

```
IMAGES OF DATA-CARDS

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 521            REACTION NO.        55          1.80E+03
 522            REACTION NO.        56          1.00E-01
 523            REACTION NO.        57           6.70E+3
 524            REACTION NO.        58          2.90E+04
 525            REACTION NO.        59          1.76E 03
 526            REACTION NO.        60          3.00E-02
 527            REACTION NO.        61          2.90E 03
 528            REACTION NO.        62          2.90E 03
 529            REACTION NO.        63          8.00E 00
 530            REACTION NO.        64          8.00E 00

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 531    1.NO2  +    HV              =  1.O     + 1.NO
 532    1.O    +    O2   +    M     =  1.O3    +   M
 533    1.O3   +  1.NO              =  1.NO2   +   O2
 534    1.NO   +  1.NO2  +  1.H2O   =  2.HONO
 535    2.HONO                      =  1.NO    + 1.NO2   + 1.H2O
 536    1.HONO +    HV              =  1.OH    + 1.NO
 537    1.OH   +  1.NO   +    M     =  1.HONO  +   M
 538    1.OH   +  1.NO2  +    M     =  1.HNO3  +   M
 539    1.OH   +  1.CO              =  1.HO2   + 1.CO2
 540    1.HO2  +  1.NO              =  1.NO2   + 1.OH

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 541    1.HO2  +  1.NO2             =  1.HNO4
 542    2.HO2                       =    H2O2  +   O2
 543    1.NO2  +  1.O3              =  1.NO3   +   O2
 544    1.NO3  +  1.NO              =  2.NO2
 545    1.NO3  +  1.NO2             =  1.N2O5
 546    1.N2O5 +  1.H2O             =  2.HNO3
 547    1.N2O5                      =  1.NO3   + 1.NO2
 548    1.OH   +  1.ALKE            =  1.AO2
 549    1.OH   +  1.C2H4            =  1.AO2
 550    1.AO2  +  1.NO              =  1.NO2   + 1.AO

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 551    1.AO   +    O2              =  .5RCHO  + 1.5HCHO + 1.HO2
 552    1.HNO4                      =  1.HO2   + 1.NO2
 553    1.O    +  1.ALKE            =  .3EPOX  + .3RCHO  + .4HO2   + .4RO2
 554    1.O    +  1.C2H4            =  .3EPOX  + .3RCHO  + .4HO2   + .4RO2   + .4CO
 555    1.O3   +  1.ALKE            =  .5HCHO  + .5RCHO  + .4HO    + .4RO    + .1RO2
 556    1.O3   +  1.C2H4            =  1.HCHO  + .8HO    + .4HO2   + .2CO2
 557    1.HO   +  1.NO              =  1.HCHO  + 1.NO2
 558    1.RO   +  1.NO              =  1.RCHO  + 1.NO2
 559    1.HO   +  1.NO2             =  1.HCHO  + 1.NO3
 560    1.RO   +  1.NO2             =  1.RCHO  + 1.NO3
```

Figure D-1 (Continued)

```
IMAGES OF DATA-CARDS

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  561   1.HO   +  1.HCHO            =     OZID
  562   1.HO   +  1.RCHO            =     OZID
  563   1.RO   +  1.HCHO            =     OZID
  564   1.RO   +  1.RCHO            =     OZID
  565   1.OH   +  1.PA              =   1.H2O  +  1.FAO2
  566   1.PAO2 +  1.NO              =   1.NO2  + .85PAO  + .15RO2
  567   1.PAO2 +  1.NO              =     NTRA
  568   1.PAO                       =   1.RO2  +  .5HCHO +  .5RCHO
  569   1.RO2  +  1.NO              =   1.NO2  +  1.PAO
  570   1.PAO  +    O2              =   1.RCHO +  1.HO2

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  571   1.PAO  +  1.NO2             =  .85NTRA + .15RCHO + .15HONO
  572   1.OH   +  1.RCHO            =   1.RCO3 +  1.H2O
  573   1.RCHO +    HV              =   1.RO2  +  1.HO2  +  1.CO
  574   1.RCO3 +  1.NO              =     CO2  +  1.NO2  +  1.RO2
  575   1.RCO3 +  1.NO2  +          =   1.PAN
  576   1.PAN                       =   1.RCO3 +  1.NO2
  577   1.HCHO +    HV              =   1.HO2  +  1.CO
  578   1.HCHO +  1.OH              =   1.H2O  +  1.HO2  +  1.CO
  579   1.RO2  +  1.HO2             =     RO2H +    O2
  580   1.OH   +  1.AP              =   1.AC   +  1.HO2

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  581   1.OH   +  1.AR              =   1.ARO
  582   1.OH   +  1.AC              =     ARP  +  1.HO2
  583   1.ARO  +  1.NO              =   1.NO2  +  1.ARIN
  584   1.ARO  +  1.NO2             =     ANO2
  585   1.ARO  +  1.HO2             =   1.AO   +  1.H2O2
  586   1.ARIN                      =   1.RO2  +  1.OH
  587   1.AR   +  1.OH              =   1.H2O  +  1.ABO
  588   1.ABO  +  1.NO              =   1.NO2  +  1.HO2  +     ACHO
  589   1.SO2  +  1.OH              =   1.SO4
  590   1.SO2  +  1.HO2             =   1.SO4  +  1.OH

      1         11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
  591   1.SO2  +  1.HO              =   1.SO4  +  1.HCHO
  592   1.SO2  +  1.RO              =   1.SO4  +  1.RCHO
  593   1.SO2  +  1.RO2             =   1.SO4  +  1.PAO
  594   1.SO2  +  1.AO2             =   1.SO4  +  1.AO
  595 LATITUDE                                     38.60
  596 LONGITUDE  ,                                 90.20
  597 TIME ZONE                                     6.00
  598 DATE                                        760629
  599 NUMBBER OF PHOTOLYTIC REACTIONS                  4
  600 REACTION NUMBER OF 1ST PHOTOLYTIC RATE           1          NO2 + HV
```

```
IMAGES OF DATA-CARDS

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  601  REACTION NUMBER OF 2ND PHOTOLYTIC RATE           5          HONO + HV
  602  REACTION NUMBER OF 3RD PHOTOLYTIC RATE          43          RCHO + HV
  603  REACTION NUMBER OF 4TH PHOTOLYTIC RATE          47          HCHO + HV
  604   CLEARNESS RATIOS                          360.000      .547
  605   CLEARNESS RATIOS                          420.000      .702
  606   CLEARNESS RATIOS                          480.000      .757
  607   CLEARNESS RATIOS                          540.000      .724
  608   CLEARNESS RATIOS                          600.000      .695
  609   CLEARNESS RATIOS                          660.000      .545
  610   CLEARNESS RATIOS                          720.000      .557

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  611   CLEARNESS RATIOS                          780.000     1.000
  612   CLEARNESS RATIOS                          840.000      .727
  613   CLEARNESS RATIOS                          900.000      .975
  614   CLEARNESS RATIOS                          960.000     1.000
  615   CLEARNESS RATIOS                         1020.000      .772
  616   CLEARNESS RATIOS                          -10.000
  617   TEMPERATURE (DEG C)                       360.000    22.051
  618   TEMPERATURE (DEG C)                       420.000    23.462
  619   TEMPERATURE (DEG C)                       480.000    25.332
  620   TEMPERATURE (DEG C)                       540.000    26.715

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  621   TEMPERATURE (DEG C)                       600.000    28.260
  622   TEMPERATURE (DEG C)                       660.000    28.693
  623   TEMPERATURE (DEG C)                       720.000    28.319
  624   TEMPERATURE (DEG C)                       780.000    28.849
  625   TEMPERATURE (DEG C)                       840.000    28.763
  626   TEMPERATURE (DEG C)                       900.000    28.947
  627   TEMPERATURE (DEG C)                       960.000    29.459
  628   TEMPERATURE (DEG C)                      1020.000    29.238
  629   TEMPERATURE (DEG C)                       -10.000
  630   DIFFUSIVITIES 1         360.0                52.3      52.3      31.4

       1         11        21        31        41        51        61        71
CARD   +.........+.........+.........+.........+.........+.........+.........+.........
  631                            20.1       8.8       8.8
  632   DIFFUSIVITIES 2         420.0                74.5      74.5      25.1
  633                            15.7       6.1       6.1
  634   DIFFUSIVITIES 3         480.0               240.2     240.2      45.1
  635                            27.5      33.0      33.0
  636   DIFFUSIVITIES 4         540.0              1331.0    1331.0     973.2
  637                            27.8      36.4      36.4
  638   DIFFUSIVITIES 5         600.0              1621.5    1621.5    3884.1
  639                          1923.0       9.0       9.0
  640   DIFFUSIVITIES 6         660.0               146.5     146.6     500.4
```

IMAGES OF DATA-CARDS

```
      1        11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 641                           290.9       8.2       8.2
 642   DIFFUSIVITIES 7         720.0              2827.8    2827.8    7543.1
 643                          4910.7     106.3     106.3
 644   DIFFUSIVITIES 8         780.0              2019.0    2019.0    6310.0
 645                          5702.1     438.4     438.4
 646   DIFFUSIVITIES 9         840.0              1729.5    1729.5    5118.5
 647                          4394.2     290.0     290.0
 648   DIFFUSIVITIES10         900.0              1424.5    1424.5    4121.3
 649                          3369.0     191.9     191.9
 650   DIFFUSIVITIES11         960.0              1439.5    1439.5    2183.3

      1        11        21        31        41        51        61        71
CARD  +.........+.........+.........+.........+.........+.........+.........+.........
 651                          1091.7      39.1      39.1
 652   DIFFUSIVITIES11        1020.0              1439.5    1439.5    2183.3
 653                          1091.7      39.1      39.1
 654   DIFFUSIVITIES11         -10.0                  .0        .0        .0
 655                              .0        .0        .0
 656  END
```

```
      SUBROUTINE DIFFUN (N,T,Y,YDOT)                                    NKM
C                                                                       NKM
C     ------      DIFFUN DEFINES THE DIFFERENTIAL EQUATIONS  ----------NKM
C                                                                       NKM
C                         YDOT = F(Y,T)                                 NKM
C                                                                       NKM
      INTEGER BCFLAG                                                    NKM
      COMMON/CHEM1/ NOSTAT,               NOSTM1,           NOREAC,     NKM
     1              NOSPEC,               NSTDY,            NK          NKM
      COMMON/CHEM2/ CONIN(40,5),          WTMOLE(40),       RATKON(65), NKM
     1              RATEFF(65),           RATEV(4,5),       QRATE,      NKM
     2              NVRATE,               LOCVRT(4)                     NKM
      COMMON/CHEM3/ ZEE(5),               DELZ(4),          HTCELL(6),  NKM
     1              TDELZ(2) ,            DFINIT(6),        SCALON(4),  NKM
     2              DCOF(5),              FLXWAL(40),       FLXDGE(40), NKM
     3              BCFLAG(40),           DPRATE(10),       DEPOWR(10), NKM
     4              LOCDPF(10),           NDPFLX ,          SCALUP(4)   NKM
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT   NKM
      COMMON /CHEM5/  REACT(20,65),SPEC(40),LOCFLX(10),                 NKM
     1                NASFLX,FLXW1(30),FLXW2(30)                        NKM
      COMMON /PS1/   TPASS(200), PS(7,5,75), FRACT(3),                  NKM
     1               NPTSR,NPSFLX,LOCPSF(7)                             NKM
      DIMENSION  Y(N),       YDOT(N),      SINK(30)                     NKM
      DATA SINK /30*0.0000/                                             NKM
C                                                                       NKM
      CALL RATES(Y,YDOT)                                                NKM
C                                                                       NKM
      IF(NDPFLX.LT.1)  GO TO 50                                         NKM
C     ----     UPDATE FLUXES OF DEPOSITING SPECIES                      NKM
      DO 40 K=1,NDPFLX                                                  NKM
      ID = LOCDPF(K)                                                    NKM
      SINK(ID) = 0.00                                                   NKM
      IF(Y(ID).LT.1.E-16)  GO TO 40                                     NKM
      SINK(ID) = - DPRATE(K)*(Y(ID)**DEPOWR(K))                         NKM
   40 CONTINUE                                                          NKM
   50 CONTINUE                                                          NKM
C                                                                       NKM
C        INTERPOLATE EMISSION FLUXES TO THE CURRENT TIME                NKM
C                                                                       NKM
      FDELT = (T - TLAST)/UPDINT                                        NKM
      DO 60 I = 1,NASFLX                                                NKM
      K = LOCFLX(I)                                                     NKM
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                  NKM
   60 CONTINUE                                                          NKM
      K = 7                                                             NKM
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                  NKM
      K = 2                                                             NKM
      FLXWAL(K) = FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                  NKM
      K = 9                                                             NKM
      FLXWAL(K)=FLXW1(K) + (FLXW2(K)-FLXW1(K))*FDELT                    NKM
      DO 75 J =1,NOSTAT                                                 NKM
      DO 70 I =1,NPSFLX                                                 NKM
      K =LOCPSF(I)                                                      NKM
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT             NKM
   70 CONTINUE                                                          NKM
      K = 2                                                             NKM
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT             NKM
      K = 7                                                             NKM
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT
      K = 9
      PSRATE(K,J) = PSR1(K,J) + (PSR2(K,J)-PSR1(K,J))*FDELT
75    CONTINUE
C
      NM1 = NK*NOSTM1
      NM2 = NK*(NOSTAT-2)
C     ------- EQUATIONS FOR SURFACE AND TOP EDGE MESH POINTS
      DO 100  I=1,NK
      IL=NM1+I
      YDOT(I) = DCOF(1)*(Y(I+NK)-Y(I)) + YDOT(I)
     1                 +  TDELZ(1)*(FLXWAL(I)+SINK(I)) + PSRATE(I,1)
      IF(BCFLAG(I) .EQ. 2)  YDOT(I) = 0.0
      YDOT(IL) = DCOF(NOSTAT)*(Y(I+NM2)-Y(IL)) + YDOT(IL)
     1                            +  PSRATE(I,NOSTAT)
  100 CONTINUE
C     ------- EQUATIONS FOR INTERMEDIATE MESH POINTS
      DO 200 J=2,NOSTM1
      NM1= (J-2)*NK
      NN = (J-1)*NK
      NP1 = J*NK
      JM1 = J- 1
      DO 150 I = 1,NK
      LN = I + NN
      YDOT(LN) = SCALOW(JM1)*Y(I+NM1) - DCOF(J)*Y(LN) +
     1           SCALUP(J)*Y(I+NP1) + YDOT(LN) + PSRATE(I,J)
  150 CONTINUE
  200 CONTINUE
      RETURN
      END


      SUBROUTINE ISTATE
C
C       COMPUTES THE INITIAL CONCENTRATIONS OF THE SPECIES LISTED BELOW
C       USING THE STEADY STATE ASSUMPTION.
C       THIS ROUTINE IS SPECIFIC TO THE ERT PHOTOCHEMICAL MECHANISM
C         39 SPECIES    X  64 REACTIONS    1.31.79
C
C             NO2      HONO     RD       HD       NO3      N205
C             ARO      AC       ARIN     ABO      AO2      AO
C             PAO2     HNO4     PAN      RCO3     RO2      PAO
C             O
C
      COMMON/CHEM1/ NOSTAT,                 NOSTM1,              NOREAC,
     1              NOSPEC,                 NSTDY,               NK
      COMMON/CHEM2/ CONIN(40,5),            WTMOLE(40),          RATKON(65),
     1              RATEFF(65),             RATEV(4,5),          QRATE,
     2              NVRATE,                 LOCVRT(4)
      DIMENSION     C(40),     R(65),       A(2,2),     B(2)
      EQUIVALENCE (R,RATEFF)
      DATA YES /3HYES /
C
      DO 200 K = 1,NOSTAT
      IF(QRATE.EQ.YES) CALL RATEHI(K)
      CALL UNMIXR(K)
      CALL XMIT(NOSPEC,CONIN(1,K),C)
```

```
C
C                  - - - NO2 - - -
      C( 2) = C( 1)*C( 3)*R( 3)/R( 1)
C                  - - -  O  - - -
      C(32) = R( 1)*C( 2)/(R( 2) + R(23)*C( 8) + R(24)*C( 9))
C                  - - - HONO - - -
      AA = 2.*R( 5)
      BB = R( 6)
      CC = -2.*R( 4)*C( 1)*C( 2)*C(33) - R( 7)*C(31)*C( 1)
      C( 4) = (-BB + (BB*BB - 4.*AA*CC)**.5)/2./AA
C                  - - -  RO - - -
      C(26) = (.40*R(25)*C( 3)*C( 8))/(R(28)*C( 1) + R(30)*C( 2)
     1          + R(33)*C( 6) + R(34)*C( 7))
C                  - - -  HO - - -
      C(25) = (.40*R(25)*C( 3)*C( 8) + .88*R(26)*C( 3)*C( 9))/
     1         (R(27)*C( 1) + R(29)*C( 2) + R(31)*C( 6) + R(32)*C( 7))
C                  - - - NO3 AND N2O5 - - -
      A(1,1) = R(14)*C( 1) + R(15)*C( 2)
      A(1,2) = -R(17)
      A(2,1) = R(15)*C( 2)
      A(2,2) = -(R(16)*C(33) + R(17))
      B( 1) = R(13)*C( 2)*C( 3) + R(29)*C( 2)*C(25) + R(30)*C( 2)*C(26)
      B( 2) = 0.0
      CALL SOL2B2(A,B)
      C(19) = B( 1)
      C(21) = B( 2)
C                  - - - ARO - - -
      C(13) = (R(51)*C(12)*C(31))/(R(53)*C( 1) + R(54)*C( 2)
     1          + R(55)*C(22))
C                  - - - AC  - - -
      C(14) = (R(50)*C(12)*C(31) + R(55)*C(13)*C(22))/(R(52)*C(31))
C                  - - - ARIN - - -
      C(15) = R(53)*C( 1)*C(13)/R(56)
C                  - - - ABO  - - -
      C(16) = R(57)*C(12)*C(31)/(R(58)*C( 1))
C                  - - - AO2  - - -
      C(18) = (R(18)*C( 8)*C(31) + R(19)*C( 9)*C(31))/(R(20)*C( 1))
C                  - - -  AO  - - -
      C(17) = R(20)*C( 1)*C(18)/R(21)
C                  - - - PAO2 - - -
      C(28) = R(35)*C(10)*C(31)/(R(36)*C( 1) + R(37)*C( 1))
C                  - - - HNO4 - - -
      C(20) = R(11)*C( 2)*C(22)/R(22)
C                  - - - PAN AND RCO3 - - -
      A(1,1) = -R(46)
      A(1,2) = R(45)*C( 2)
      A(2,1) = +R(46)
      A(2,2) = -R(44)*C( 1) - R(45)*C( 2)
      B( 1) = 0.0
      B( 2) = -R(42)*C( 7)*C(31)
      CALL SOL2B2(A,B)
      C(11) = B(1)
      C(24) = B(2)
C                  - - - RO2 AND PAO - - -
      A(1,1) = R(39)*C( 1) + R(49)*C(22)
      A(1,2) = -R(38)
      A(2,1) = -R(39)*C( 1)
      A(2,2) = R(38) + R(40) + R(41)*C( 2)
      B(1) = .40*R(23)*C( 8)*C(32) + .10*R(25)*C( 3)*C( 8) + .15*R( 36)*NKM  172
     1        C( 1)*C(28) + R(43)*C( 7) + R(44)*C( 1)*C(24) + R(56)*C(15)NKM  173
     2  + 0.4*R(24)*C(9)*C(32)                                          NKM  174
      B(2) = .85*R(36)*C( 1)*C(28)                                      NKM  175
      CALL SOL292(A,B)                                                  NKM  176
      C(23) = B(1)                                                      NKM  177
      C(27) = B(2)                                                      NKM  178
      CALL XMIT(NOSPEC,C,CONIN(1,K))                                    NKM  179
200   CONTINUE                                                          NKM  180
      RETURN                                                            NKM  181
      END                                                               NKM  182



      SUBROUTINE JACOB(A,C,R,N)                                         NKM  183
C                                                                       NKM  184
C     JACOB COMPUTES THE JACOBIAN OF THE CHEMICAL RATE EQUATIONS.       NKM  185
C                                                                       NKM  186
      DIMENSION    A(N,N),      C(1),      R(1)                         NKM  187
C                                                                       NKM  188
C     ZERO THE A ARRAY                                                  NKM  189
      NN2 = N*N                                                         NKM  190
      CALL XMIT(-NN2,0.0,A)                                             NKM  191
C                                                                       NKM  192
C   JACOBIAN OF THE ERT PHOTOCHEMICAL MECHANISM  (39X64)  1.31.79       NKM  193
C                                                                       NKM  194
      A(  1,  1) = - R(  3)*C(  3) - R(  4)*C(  2)*C( 33) - R(  7)      NKM  195
     *     *C( 31) - R( 10)*C( 22) - R( 14)*C( 19) - R( 20)*C( 18) -    NKM  196
     *     R( 27)*C( 25) - R( 28)*C( 26) - R( 36)*C( 28) - R( 37)*C( 28) NKM  197
     *      - R( 39)*C( 23) - R( 44)*C( 24) - R( 53)*C( 13) - R( 58)    NKM  198
     *     *C( 16)                                                      NKM  199
      A(  1,  2) = + R(  1) - R(  4)*C(  1)*C( 33)                      NKM  200
      A(  1,  3) = - R(  3)*C(  1)                                      NKM  201
      A(  1,  4) = +    2.00*R(  5)*C(  4) + R(  6)                     NKM  202
      A(  1, 13) = - R( 53)*C(  1)                                      NKM  203
      A(  1, 16) = - R( 58)*C(  1)                                      NKM  204
      A(  1, 18) = - R( 20)*C(  1)                                      NKM  205
      A(  1, 19) = - R( 14)*C(  1)                                      NKM  206
      A(  1, 22) = - R( 10)*C(  1)                                      NKM  207
      A(  1, 23) = - R( 39)*C(  1)                                      NKM  208
      A(  1, 24) = - R( 44)*C(  1)                                      NKM  209
      A(  1, 25) = - R( 27)*C(  1)                                      NKM  210
      A(  1, 26) = - R( 28)*C(  1)                                      NKM  211
      A(  1, 28) = - R( 36)*C(  1) - R( 37)*C(  1)                      NKM  212
      A(  2,  1) = + R(  3)*C(  3) - R(  4)*C(  2)*C( 33) + R( 10)      NKM  213
     *     *C( 22) +    2.00*R( 14)*C( 19) + R( 20)*C( 18) + R( 27)     NKM  214
     *     *C( 25) + R( 28)*C( 26) + R( 36)*C( 28) + R( 39)*C( 23) +    NKM  215
     *     R( 44)*C( 24) + R( 53)*C( 13) + R( 58)*C( 16)                NKM  216
      A(  2,  2) = - R(  1) - R(  4)*C(  1)*C( 33) - R(  8)*C( 31) -    NKM  217
     *     R( 11)*C( 22) - R( 13)*C(  3) - R( 15)*C( 19) - R( 29)*C( 25) NKM  218
     *      - R( 30)*C( 26) - R( 41)*C( 27) - R( 45)*C( 24) - R( 54)    NKM  219
     *     *C( 13)                                                      NKM  220
      A(  2,  3) = + R(  3)*C(  1) - R( 13)*C(  2)                      NKM  221
      A(  2,  4) = +    2.00*R(  5)*C(  4)                              NKM  222
      A(  2, 11) = + R( 46)                                             NKM  223
      A(  2, 13) = + R( 53)*C(  1) - R( 54)*C(  2)                      NKM  224
      A(  2, 16) = + R( 58)*C(  1)                                      NKM  225
      A(  2, 18) = + R( 20)*C(  1)                                      NKM  226
      A(  2, 19) = +    2.00*R( 14)*C(  1) - R( 15)*C(  2)
      A(  2, 20) = + R( 22)                                             NKM  227
      A(  2, 21) = + R( 17)                                             NKM  228
      A(  2, 22) = + R( 10)*C(  1) - R( 11)*C(  2)                      NKM  229
      A(  2, 23) = + R( 39)*C(  1)                                      NKM  230
      A(  2, 24) = + R( 44)*C(  1) - R( 45)*C(  2)                      NKM  231
      A(  2, 25) = + R( 27)*C(  1) - R( 29)*C(  2)                      NKM  232
      A(  2, 26) = + R( 28)*C(  1) - R( 30)*C(  2)                      NKM  233
      A(  2, 27) = - R( 41)*C(  2)                                      NKM  234
      A(  2, 28) = + R( 36)*C(  1)                                      NKM  235
      A(  3,  1) = - R(  3)*C(  3)                                      NKM  236
      A(  3,  2) = - R( 13)*C(  3)                                      NKM  237
      A(  3,  3) = - R(  3)*C(  1) - R( 13)*C(  2) - R( 25)*C(  8) -    NKM  238
     *     R( 26)*C(  9)                                                NKM  239
      A(  3,  8) = - R( 25)*C(  3)                                      NKM  240
      A(  3,  9) = - R( 26)*C(  3)                                      NKM  241
      A(  4,  1) = +    2.00*R(  4)*C(  2)*C( 33) + R(  7)*C( 31)       NKM  242
      A(  4,  2) = +    2.00*R(  4)*C(  1)*C( 33) +    0.15*R( 41)      NKM  243
     *     *C( 27)                                                      NKM  244
      A(  4,  4) = -    4.00*R(  5)*C(  4) - R(  6)                     NKM  245
      A(  4, 27) = +    0.15*R( 41)*C(  2)                              NKM  246
      A(  5,  5) = - R(  9)*C( 31)                                      NKM  247
      A(  5,  6) = + R( 47) + R( 48)*C( 31)                             NKM  248
      A(  5,  7) = + R( 43)                                             NKM  249
      A(  5,  9) = +    0.40*R( 24)*C( 32)                              NKM  250
      A(  6,  1) = + R( 27)*C( 25)                                      NKM  251
      A(  6,  2) = + R( 29)*C( 25)                                      NKM  252
      A(  6,  3) = +    0.50*R( 25)*C(  8) + R( 26)*C(  9)              NKM  253
      A(  6,  6) = - R( 31)*C( 25) - R( 33)*C( 26) - R( 47) - R( 48)    NKM  254
     *     *C( 31)                                                      NKM  255
      A(  6,  8) = +    0.50*R( 25)*C(  3)                              NKM  256
      A(  6,  9) = + R( 26)*C(  3)                                      NKM  257
      A(  6, 17) = +    1.50*R( 21)                                     NKM  258
      A(  6, 25) = + R( 27)*C(  1) + R( 29)*C(  2) - R( 31)*C(  6) +    NKM  259
     *     R( 61)*C( 29)                                                NKM  260
      A(  6, 26) = - R( 33)*C(  6)                                      NKM  261
      A(  6, 27) = +    0.50*R( 38)                                     NKM  262
      A(  7,  1) = + R( 28)*C( 26)                                      NKM  263
      A(  7,  2) = + R( 30)*C( 26) +    0.15*R( 41)*C( 27)              NKM  264
      A(  7,  3) = +    0.50*R( 25)*C(  8)                              NKM  265
      A(  7,  7) = - R( 32)*C( 25) - R( 34)*C( 26) - R( 42)*C( 31) -    NKM  266
     *     R( 43)                                                       NKM  267
      A(  7,  8) = +    0.30*R( 23)*C( 32) +    0.50*R( 25)*C(  3)      NKM  268
      A(  7,  9) = +    0.30*R( 24)*C( 32)                              NKM  269
      A(  7, 17) = +    0.50*R( 21)                                     NKM  270
      A(  7, 25) = - R( 32)*C(  7)                                      NKM  271
      A(  7, 26) = + R( 28)*C(  1) + R( 30)*C(  2) - R( 34)*C(  7) +    NKM  272
     *     R( 62)*C( 29)                                                NKM  273
      A(  7, 27) = +    0.50*R( 38) + R( 40) +    0.15*R( 41)*C(  2)    NKM  274
      A(  8,  3) = - R( 25)*C(  8)                                      NKM  275
      A(  8,  8) = - R( 18)*C( 31) - R( 23)*C( 32) - R( 25)*C(  3)      NKM  276
      A(  9,  3) = - R( 26)*C(  9)                                      NKM  277
      A(  9,  9) = - R( 19)*C( 31) - R( 24)*C( 32) - R( 26)*C(  3)      NKM  278
      A( 10, 10) = - R( 35)*C( 31)                                      NKM  279
      A( 11,  2) = + R( 45)*C( 24)                                      NKM  280
      A( 11, 11) = - R( 46)                                             NKM  281
      A( 11, 24) = + R( 45)*C(  2)                                      NKM  282
      A( 12, 12) = - R( 50)*C( 31) - R( 51)*C( 31) - R( 57)*C( 31)      NKM  283
                                                                        NKM  284
      A( 13,  1) = - R( 53)*C( 13)                                      NKM  285
      A( 13,  2) = - R( 54)*C( 13)                                      NKM  286
      A( 13, 12) = + R( 51)*C( 31)                                      NKM  287
      A( 13, 13) = - R( 53)*C(  1) - R( 54)*C(  2) - R( 55)*C( 22)      NKM  288
      A( 13, 22) = - R( 55)*C( 13)                                      NKM  289
      A( 14, 12) = + R( 50)*C( 31)                                      NKM  290
      A( 14, 13) = + R( 55)*C( 22)                                      NKM  291
      A( 14, 14) = - R( 52)*C( 31)                                      NKM  292
      A( 14, 22) = + R( 55)*C( 13)                                      NKM  293
      A( 15,  1) = + R( 53)*C( 13)                                      NKM  294
      A( 15, 13) = + R( 53)*C(  1)                                      NKM  295
      A( 15, 15) = - R( 56)                                             NKM  296
      A( 16,  1) = - R( 58)*C( 16)                                      NKM  297
      A( 16, 12) = + R( 57)*C( 31)                                      NKM  298
      A( 16, 16) = - R( 58)*C(  1)                                      NKM  299
      A( 17,  1) = + R( 20)*C( 18)                                      NKM  300
      A( 17, 17) = - R( 21)                                             NKM  301
      A( 17, 18) = + R( 20)*C(  1) + R( 64)*C( 29)                      NKM  302
      A( 18,  1) = - R( 20)*C( 18)                                      NKM  303
      A( 18,  8) = + R( 18)*C( 31)                                      NKM  304
      A( 18,  9) = + R( 19)*C( 31)                                      NKM  305
      A( 18, 18) = - R( 20)*C(  1) - R( 64)*C( 29)                      NKM  306
      A( 19,  1) = - R( 14)*C( 19)                                      NKM  307
      A( 19,  2) = + R( 13)*C(  3) - R( 15)*C( 19) + R( 29)*C( 25) +    NKM  308
     *   R( 30)*C( 26)                                                  NKM  309
      A( 19,  3) = + R( 13)*C(  2)                                      NKM  310
      A( 19, 19) = - R( 14)*C(  1) - R( 15)*C(  2)                      NKM  311
      A( 19, 21) = + R( 17)                                             NKM  312
      A( 19, 25) = + R( 29)*C(  2)                                      NKM  313
      A( 19, 26) = + R( 30)*C(  2)                                      NKM  314
      A( 20,  2) = + R( 11)*C( 22)                                      NKM  315
      A( 20, 20) = - R( 22)                                             NKM  316
      A( 20, 22) = + R( 11)*C(  2)                                      NKM  317
      A( 21,  2) = + R( 15)*C( 19)                                      NKM  318
      A( 21, 19) = + R( 15)*C(  2)                                      NKM  319
      A( 21, 21) = - R( 16)*C( 33) - R( 17)                             NKM  320
      A( 22,  1) = - R( 10)*C( 22) + R( 58)*C( 16)                      NKM  321
      A( 22,  2) = - R( 11)*C( 22)                                      NKM  322
      A( 22,  3) = +     0.40*R( 26)*C(  9)                             NKM  323
      A( 22,  5) = + R(  9)*C( 31)                                      NKM  324
      A( 22,  6) =                R( 47) + R( 48)*C( 31)                NKM  325
      A( 22,  7) = + R( 43)                                             NKM  326
      A( 22,  8) = +     0.40*R( 23)*C( 32)                             NKM  327
      A( 22,  9) = +     0.40*R( 24)*C( 32) +     0.40*R( 26)*C(  3)    NKM  328
      A( 22, 12) = + R( 50)*C( 31)                                      NKM  329
      A( 22, 13) = - R( 55)*C( 22)                                      NKM  330
      A( 22, 14) = + R( 52)*C( 31)                                      NKM  331
      A( 22, 16) = + R( 58)*C(  1)                                      NKM  332
      A( 22, 17) = + R( 21)                                             NKM  333
      A( 22, 20) = + R( 22)                                             NKM  334
      A( 22, 22) = - R( 10)*C(  1) - R( 11)*C(  2) -     4.00*R( 12)    NKM  335
     *   *C( 22) - R( 49)*C( 23) - R( 55)*C( 13) - R( 60)*C( 29)        NKM  336
      A( 22, 23) = - R( 49)*C( 22)                                      NKM  337
      A( 22, 27) = + R( 40)                                             NKM  338
      A( 23,  1) = +     0.15*R( 36)*C( 28) - R( 39)*C( 23) + R( 44)    NKM  339
     *   *C( 24)                                                        NKM  340
      A( 23,  3) = +     0.10*R( 25)*C(  8)                             NKM  341
      A( 23,  7) = + R( 43)                                             NKM  342
      A( 23,  8) = +    0.40*R( 23)*C( 32) +    0.10*R( 25)*C(  3)
      A( 23,  9) = +    0.40*R( 24)*C( 32)
      A( 23, 15) = + R( 56)
      A( 23, 22) = - R( 49)*C( 23)
      A( 23, 23) = - R( 39)*C(  1) - R( 49)*C( 22) - R( 63)*C( 29)
      A( 23, 24) = + R( 44)*C(  1)
      A( 23, 27) = + R( 38)
      A( 23, 28) = +    0.15*R( 36)*C(  1)
      A( 24,  1) = - R( 44)*C( 24)
      A( 24,  2) = - R( 45)*C( 24)
      A( 24,  7) = + R( 42)*C( 31)
      A( 24, 11) = + R( 46)
      A( 24, 24) = - R( 44)*C(  1) - R( 45)*C(  2)
      A( 25,  1) = - R( 27)*C( 25)
      A( 25,  2) = - R( 29)*C( 25)
      A( 25,  3) = +    0.40*R( 25)*C(  8) +    0.80*R( 26)*C(  9)
      A( 25,  6) = - R( 31)*C( 25)
      A( 25,  7) = - R( 32)*C( 25)
      A( 25,  8) = +    0.40*R( 25)*C(  3)
      A( 25,  9) = +    0.80*R( 26)*C(  3)
      A( 25, 25) = - R( 27)*C(  1) - R( 29)*C(  2) - R( 31)*C(  6) -
     *    R( 32)*C(  7) - R( 61)*C( 29)
      A( 26,  1) = - R( 28)*C( 26)
      A( 26,  2) = - R( 30)*C( 26)
      A( 26,  3) = +    0.40*R( 25)*C(  8)
      A( 26,  6) = - R( 33)*C( 26)
      A( 26,  7) = - R( 34)*C( 26)
      A( 26,  8) = +    0.40*R( 25)*C(  3)
      A( 26, 26) = - R( 28)*C(  1) - R( 30)*C(  2) - R( 33)*C(  6) -
     *    R( 34)*C(  7) - R( 62)*C( 29)
      A( 27,  1) = +    0.85*R( 36)*C( 28) + R( 39)*C( 23)
      A( 27,  2) = - R( 41)*C( 27)
      A( 27, 23) = + R( 39)*C(  1) + R( 63)*C( 29)
      A( 27, 27) = - R( 38) - R( 40) - R( 41)*C(  2)
      A( 27, 28) = +    0.85*R( 36)*C(  1)
      A( 28,  1) = - R( 36)*C( 28) - R( 37)*C( 28)
      A( 28, 10) = + R( 35)*C( 31)
      A( 28, 28) = - R( 36)*C(  1) - R( 37)*C(  1)
C
      IF(N.LE.28)    RETURN
C
C
C         ELEMENTS OF THE JACOBIAN FOR SO2 AND SO4 REACTIONS
C
      A(  6, 29) = + R( 61)*C( 25)
      A(  7, 29) = + R( 62)*C( 26)
      A( 17, 29) = + R( 64)*C( 18)
      A( 18, 29) = - R( 64)*C( 18)
      A( 22, 29) = - R( 60)*C( 22)
      A( 23, 29) = - R( 63)*C( 23)
      A( 25, 29) = - R( 61)*C( 25)
      A( 26, 29) = - R( 62)*C( 26)
      A( 27, 29) = + R( 63)*C( 23)
      A( 29, 18) = - R( 64)*C( 29)
      A( 29, 22) = - R( 60)*C( 29)
      A( 29, 23) = - R( 63)*C( 29)
      A( 29, 25) = - R( 61)*C( 29)
      A( 29, 26) = - R( 62)*C( 29)
      A( 29, 29) = - R( 59)*C( 31) - R( 60)*C( 22) - R( 61)*C( 25) -
     *    R( 62)*C( 26) - R( 63)*C( 23) - R( 64)*C( 18)
      A( 30, 18) = + R( 64)*C( 29)
      A( 30, 22) = + R( 60)*C( 29)
      A( 30, 23) = + R( 63)*C( 29)
      A( 30, 25) = + R( 61)*C( 29)
      A( 30, 26) = + R( 62)*C( 29)
      A( 30, 29) = + R( 59)*C( 31) + R( 60)*C( 22) + R( 61)*C( 25) +
     *    R( 62)*C( 26) + R( 63)*C( 23) + R( 64)*C( 18)
C
      RETURN
      END



      SUBROUTINE KEMOD2(INTIM,TSTOP,NUMFLX,FLXUNT)
C
C     -------  KEMOD2 IS THE SECONDARY DRIVER FOR THE CHEMICAL/
C              DIFFUSION MODULE OF THE ERT TRAJECTORY MODEL.
C
C              KEMOD2 CONTROLS ALL I/O EXCEPT INITIAL EMISSION INPUTS.
C              KEMOD2 CONTROLS THE UPDATING PROCESSES.
C
C     -------  FEATURES OF KEMOD2
C                  INTEGRATION BY EPISODE PACKAGE
C                  VARIABLE SIZE VERTICAL MESH
C                  GENERALIZED BOUNDARY CONDITIONS
C                  TIME + HEIGHT DEPENDENT PHOTODISSOCIATION RATES,
C                   WITH SKY CLEARNESS RATIOS
C                  TEMPERATURE DEPENDENT REACTION RATE CONSTANTS
C
      REAL INTIM
      INTEGER BCFLAG
      COMMON/CHEM1/ NOSTAT,               NOSTM1,               NOREAC,
     1              NOSPEC,               NSTDY,                NK
      COMMON/CHEM2/ CONIN(40,5),          WTHOLE(40),           RATKON(65),
     1              RATEFF(65),           RATEV(4,5),           QRATE,
     2              NVRATE,               LOCVRT(4)
      COMMON/CHEM3/ ZEE(5),               DELZ(4),              HTCELL(6),
     1              TDELZ(2) ,            DFINIT(6),            SCALOW(4),
     2              DCOF(5),              FLXWAL(40),           FLXDGE(40),
     3              BCFLAG(40),           DPRATE(10),           DEPOWR(10),
     4              LOCDPF(10),           NDPFLX ,              SCALUP(4)
      COMMON /CHEM4/  RLAT,  RLONG,  TMZONE,  HIRATE,  JDATE
      COMMON /CHEM5/  REACT(20,65),SPEC(40),LOCFLX(10),
     1              NASFLX,FLXW1(30),FLXW2(30)
      COMMON/INPUTS/TITLE(20),            IDATE(10),            NCURV
      COMMON/FLUXES/FLXIN( 7,200),        FLXTIM(200),          NFLUX
      COMMON /PS1/  TPASS(200),           PS(7,5,75),           FRACT(3),
     1              NPTSR,                NPSFLX,               LOCPSF(7)
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT
      COMMON/EPCOM6/PW(4500)
      COMMON/PRPLOT/KPS(5) ,              KSYM(5),              GRCON(5,100)
     1    ,       TOUT(100),    VALMAX(5)
      COMMON/EPCOM9/HUSED,                NQUSED,               NSTEP,
     1              NFE,                  NJE
      DIMENSION     YO(150),              PPM (40,5),           TEMP(25),
     1              TMTEMP(25)  ,         HTINV(100,8),         WORK(200),
     2                CLOUDT(25),          CLOUDF(25)          ,VFR(5,200),
     3                PTSR(7,200)
C
      EQUIVALENCE (YES,IES),  (RNEG,NEG)
      EQUIVALENCE (PPM ,CONIN)
      EQUIVALENCE (VFR,PHI),  (PTSR,PW(1001))
C
      DATA LUP, LIN, LOUT, YES, RNEG  /1, 3, 6, 3HYES, 2HNO/
      DATA KSYM /1HN, 1H2, 1HO, 1HS, 1HC  /
      DATA RMOLE /4HMOLE/
      DATA NROW,MAXR,MAXMSH,MAXNK,NRHT,KTMAX /40,55,5,30,100,100/
      DATA MF,IERROR /21,3/
C
C------------ READ INPUTS FOR COMPUTING POLLUTANT CONCENTRATIONS
C
      READ(LIN,30) DELT
      READ(LIN,30) BIGSTP
      READ(LIN,30) EPS
      READ(LIN,30) UPDINT
      READ(LIN,30) PRNTIN
      READ(LIN,29) TIMOUT
      READ(LIN,29) SUNGEN
      READ(LIN,29) HIRATE
      READ(LIN,29) OCLOUD
      READ(LIN,29) VTEMP
      READ(LIN,29) SPUNCH
      READ(LIN,31) NUMSTP
      READ(LIN,31) NOREAC
      READ(LIN,31) NOSPEC
      READ(LIN,31) NSTDY
      READ(LIN,31) NOSTAT
      READ(LIN,30) (ZEE(I), I=1,NOSTAT)
C
      NK = NOSPEC -NSTDY
      NPUNCH = NK
      SUNTIM = UPDINT
      ROCHEM = YES
      ORATE = YES
      F = YES
      TSTOP2 = TSTOP
      KOK = 100
      NASFLX = NUMFLX
C
C     ---- NOSPEC IS THE TOTAL NUMBER OF SPECIE    (MAX = 40)
C     ---- NSTDY IS THE NUMBER OF SPECIE HELD CONSTANT OR IN STEADY STAT
C     ---- NK IS THE NUMBER OF SPECIE INTEGRATED    (MAXNK = 30)
C     ---- NOSTAT IS THE NUMBER OF VERTICAL STATIONS (MIN=4,MAX=5)
C     -----LOCFLX IS LOCATION  INDEX FOR AREA SOURCE EMISSION FLUXES
C     -----NUMFLX IS NUMBER OF AREA SOURCE EMISSION FLUXES
C     -----LOCPSF IS LOCATION  INDEX OF POINT SOURCE EMISSION FLUXES
C     -----NPSFLX IS NUMBER OF POINT SOURCE EMMISION FLUXES
C     -----LOCDPF IS LOCATION  INDEX FOR DEPOSITING FLUXES
C     -----NDPFLX IS NUMBER OF DEPOSITING FLUXES
C
C     --------------BOUNDARY CONDITION CODE FOR WALL--------------------
C
C     BCFLAG = 0   DC/DZ = PHI/KZ   WHERE    PHI = 0.  (DEFAULT CASE)
C                                   OR       PHI = AREA SOURCE EMISSION F
```

```
C
C                                           AND     KZ  = DIFFUSION COEFFICIENT
C        BCFLAG = 1    DC/DZ = -DPRATE*(PPM**DEPOWR)/KZ  DEPOSITING SPECIE
C        -----WHERE
C        -----DPRATE(I) IS THE DEPOSITION VELOCITY OF THE I*TH  SPECIE
C        -----IN METERS/MINUTE/(PPM**(DEPOWR-1))
C        -----INPUT AS POSITIVE QUANTITY FOR UPTAKE AT WALL
C
C        BCFLAG = 2    C = INITIAL CONCENTRATION FOR ALL TIME
C
C
C        ----------------------------------------------------------------------
C        ------  ZERO PPM ARRAY  AND CHECK NK AND NOSTAT INPUTS
      NRC = MAXMSH*NROW
      CALL XMIT (-NRC,0.,CONIN)
      IF(NOSTAT.GE.4)  GO TO 200
      WRITE(LOUT,77) NOSTAT
      GO TO 800
  200 CONTINUE
      IF(NK.LE.MAXNK)  GO TO 210
      WRITE(LOUT,76) NOSPEC ,NSTDY, NK
      GO TO 800
  210 CONTINUE
      IF(ROCHEM.NE.YES)  GO TO 219
      NDPFLX = 0
      DO 215  I=1,NCSPEC
      READ(LIN,51)  SPEC(I),WTMOLE(I),BCFLAG(I)
      IF(BCFLAG(I) .EQ. 1)  NDPFLX = NDPFLX + 1
  215 CONTINJE
      IF(NUMFLX.GT.0) READ(LIN,31) (LOCFLX(I),I=1,NUMFLX)
      IF(NPSFLX.GT.0) READ(LIN,31) (LOCPSF(I),I=1,NPSFLX)
      IF(NDPFLX.GT.0) READ(LIN,53) (LOCDPF(I),DPRATE(I),DEPOWR(I),
     1  I=1,NDPFLX)
      IF(NDPFLX.LE.0)  GO TO 219
C     ------ ADJUST DEPOSITION VELOCITIES SO AS TO EFFECT 10 METER LAYER
      A = .5*(ZEE(2)-ZEE(1))
      A = 10./A
      DO 218 I = 1,NDPFLX
  218 DPRATE(I) = A*DPRATE(I)
  219 CONTINUE
      DO 220 I = 1,NOSPEC
      READ(LIN,41)  (PPM(I,J),J=1,NOSTAT)
  220 CONTINUE
      IF(ROCHEM.NE.YES)  GO TO 230
      DO 222 I=1,NOREAC
      READ(LIN,30) RATKON(I)
  222 CONTINUE
      DO 224 J=1,NOREAC
      READ(LIN,40)  (REACT(I,J),I=1,20)
  224 CONTINUE
C
  230 CONTINUE
C
      TIME = INTIM
C
C     ------- THE PHOTODISSOCIATION RATES VARY WITH HEIGHT IF HIRATE=YES
C     ------- LOCVRT IS THE SPECIES INDEX FOR THE RATES (INPUT)
C
```

```
      READ(LIN,30) RLAT
      READ(LIN,30) RLONG
      READ(LIN,30) THZONE
      READ(LIN,31) JDATE
      READ(LIN,31)  NVRATE
      READ(LIN,31) (LOCVRT(I),I=1,NVRATE)
C
      IF(SUNGEN.EQ.YES)  GO TO 244
      WRITE(LOUT,22)
      STOP
  244 CONTINUE
C
      IF(QCLOUD.EQ.RNEG)  GO TO 249
C     ------- INPUT SKY COVER FACTORS AND TIMES
C     ------- LAST SKY COVER UPDATE TIME (CLOUDT) MUST BE A NEGATIVE
      DO 246  I=1,26
      READ(LIN,59)  A, B
      IF(A.LT.0.0)  GO TO 248
      IF(I.GT.25)  GO TO 247
      CLOUDT(I) = A
      CLOUDF(I) = B
      NCLOUD = I
  246 CONTINUE
      GO TO 248
  247 WRITE(LOUT,57)
      GO TO 800
  248 CONTINUE
      ICLOUD = 1
      CLOUDY = CLOUDF(1)
  249 IF(QCLOUD.EQ.RNEG)  CLOUDY= 1.0
C     ------   CALCULATE INITIAL PHOTODISSOCIATION RATES
      CALL PHOTOR(TIME,NOSTAT,ZEE,CLOUDY,RATEV)
      CALL RATEHI(1)
260   CONTINUE
C
C     ------------ READ TEMPERATURES IF VTEMP = YES
      ITEMP = 1
      NTEMP = 1
      TEMP(1) = 298.2
      IF(VTEMP.NE.YES)  GO TO 265
      DO 261 I = 1,26
      READ(LIN,59)  A, B
      IF(I.GT.25)   GO TO 262
      IF(A.LT.0.)  GO TO 265
      THTEMP(I) = A
      TEMP(I) = B + 273.2
      NTEMP = I
  261 CONTINUE
  262 CONTINUE
      WRITE(LOUT,79)
      GO TO 800
  265 CONTINUE
      CALL TEMPR (ITEMP,INTIM,TEMP)
C
C     -------- READ TABLE OF DIFFUSIVITY UPDATE TIMES, MIXING HEIGHTS,
C              AND DIFFUSIVITY COEFFICIENTS.
C              MIXING HEIGHT INFORMATION IS OPTIONAL.
      K = NOSTAT+3
      DO 270 I = 1,101
      READ(LIN,41) A,B,(HTINV(I,J),J=3,K)
      IF (A.LT.0.) GO TO 280
      HTINV(I,1) = A
      HTINV(I,2) = B
  270 CONTINUE
      PRINT 45
      GO TO 800
  280 NHT = I-1
      IHT = 1
      KHT = 1
  290 CONTINUE
C
C     INITIALIZE PLOT PARAMETERS
C      FOR
      KPS(1) = 1
      KPS(2) = 2
      KPS(3) = 3
      KPS(4) = 29
      KPS(5) = 5
      NCURV = 5
      CALL XMIT(-NCURV,0.0,VALMAX)
C
C      BANNER PAGE
C
      WRITE(LOUT,99)
C
C     ----- REGULATE THE AREA SOURCE EMISSION FLUX SCHEDULES
      IF(NUMFLX.EQ.0)  GO TO 299
      DO 297 I=1,NUMFLX
      DO 293 J=1,NFLUX
      WORK(J) = FLXIN(I,J)
  293 CONTINUE
      CALL SKEDUL(WORK,FLXTIM,NFLUX,NUPDAT,UPDINT,INTIM,TSTOP,-1.,F,-1)
      IF(F.E2. YES)  GO TO 800
C     -----    CONVERT FLUXES FROM MASS OR MOLE FRACTION TO PPM
      ID = LOCFLX(I)
      WT = 1.E6
      IF(FLXUNT.NE.RMOLE)    WT = 28.97E6 /WTMOLE(ID)
      DO 295 J=1,NUPDAT
      FLXIN(I,J) = WORK(J)*WT
  295 CONTINUE
  297 CONTINUE
C     -----------  THE NEW SCHEDULE OF FLUXES IS STORED IN FLXIN
      NFLUX = NUPDAT
      IFLXTM = 1
  299 CONTINUE
C     ------- INITIALIZE FLUXES
      NPMX = 5*MAXNK
      CALL  XMIT(-NPMX,0.0,PSR1)
      CALL  XMIT(-NPMX,0.0,PSR2)
      CALL  XMIT(-MAXNK,0.0,FLXW1)
      CALL  XMIT(-MAXNK,0.0,FLXW2)
      CALL XMIT(-NROW,0.,FLXWAL)
      CALL XMIT(-NROW,0.,FLXDGE)
      CALL UPFLX1(TIME,IFLXTM)
C
C     -----   GENERATE VERTICAL MESH PARAMETERS
```

```
C
      NOSTM1 = NOSTAT - 1
      DELZ(1) = ZEE(2) - ZEE(1)
      HTCELL(1) = DELZ(1)
      DO 300 K = 2,NOSTM1
      DELZ(K) = ZEE(K+1) - ZEE(K)
      HTCELL(K) = .5*(DELZ(K) + DELZ(K-1))
300   CONTINUE
      HTCELL(NOSTAT) = DELZ(NOSTM1)
C
C      ----- CONVERT POINT SOURCE EMISSIONS FROM ABSOLUTE MASS
C            OR MOLES TO RATES ON A REGULAR UPDATE SCHEDULE.
C
C      ----- POINT SOURCE MASS IS BLED IN OVER (UP TO) THREE
C            UPDATE INTERVALS. THE FRACTION IN EACH INTERVAL
C            IS SET BY THE FRACT ARRAY (SEE KEMOD DATA STATEMENT).
C
      IF(NPSFLX.LE.0)  GO TO 307
      CALL XMIT(-2625,0.0,PS)
      A = 1.E6/UPDINT
      IF(FLXUNT.EQ.RMOLE)  GO TO 303
      DO 302 K = 1,NPSFLX
      ID =LOCPSF(K)
      WORK(K) = 28.97/WTMOLE(ID)
302   CONTINUE
303   CONTINUE
      B = INTIM - 0.499*UPDINT
      K4 = 0
      DO 305 K = 1,NPTSR
      K1 = IFIX( (TPASS(K)-B)/UPDINT) + 1
      K2 = K1 + 1
      K3 = K1 + 2
      K4 = MAX0(K4,K3)
      DO 305 J = 1,NOSTAT
      VF = A*VFR(J,K)/HTCELL(J)
      IF(J.EQ.1  .OR.  J.EQ.NOSTAT)  VF = 2.*VF
      DO 305 I = 1,NPSFLX
      C = VF*PTSR(I,K)
      IF(FLXUNT.NE.RMOLE) C = C*WORK(I)
      PS(I,J,K1) = PS(I,J,K1) + FRACT(1)*C
      PS(I,J,K2) = PS(I,J,K2) + FRACT(2)*C
      PS(I,J,K3) = PS(I,J,K3) + FRACT(3)*C
305   CONTINUE
      NPTSR = K4
      TPASS(1) = INTIM
      DO 306 K = 2,NPTSR
      TPASS(K) = TPASS(K-1) + UPDINT
306   CONTINUE
      N = MAXNK*NOSTAT
      CALL XMIT(-N,0.0,PSRATE)
      CALL XMIT(-2400,0.0,PW)
307   IPS = 1
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)
C
C     ------- PRINT INPUTS
C
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,16) INTIM,TSTOP,DELT,BIGSTP,NOREAC,NOSPEC,NSTDY,NOSTAT
      IF(NUMFLX.EQ.0)  GO TO 311
      DO 308 I= 1,NUMFLX
      ID = LOCFLX(I)
      WORK(I) = SPEC(ID)
  308 CONTINUE
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,68)  (WORK(I) , I=1,NUMFLX)
      DO 310  J = 1, NFLUX
      IF(MOD(J,47).NE.0)  GO TO 309
      CALL NEWPAG (TITLE,0,IDATE)
      WRITE(LOUT,68)  (WORK(I), I=1,NUMFLX)
  309 WRITE(LOUT,69) FLXTIM(J), (FLXIN(I,J), I=1,NUMFLX)
      FLXTIM(J+1)=FLXTIM(J) + UPDINT
  310 CONTINUE
  311 CONTINUE
      IF(NPSFLX.EQ.0)  GO TO 317
      DO 312  K=1,NPSFLX
      ID = LOCPSF(K)
      WORK(K) = SPEC(ID)
  312 CONTINUE
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,83)  (WORK(K),K=1,NPSFLX)
      DO 314 K = 1,NPTSR
      IF(MOD(K,8).NE.0)  GO TO 313
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,83)  (WORK(KK),KK=1,NPSFLX)
  313 WRITE(LOUT,84)  TPASS(K), (PS(I,1,K),I=1,NPSFLX)
      DO 314 J =2,NOSTAT
      WRITE(LOUT,85)  (PS(I,J,K),I=1,NPSFLX)
  314 CONTINUE
  317 CONTINUE
C
      CALL XMIT(NOREAC,RATKON,RATEFF)
      CALL UNMIXR(1)
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,37)
      DO 320 I=1,NOREAC
      IF(MOD(I,44).NE.0)  GO TO 319
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,37)
  319 WRITE(LOUT,38) I, (REACT(J,I),J=1,20), RATEFF(I)
  320 CONTINUE
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,28)
      DO 330 J = 1,NOSPEC
  325 WRITE(LOUT,27)  J,  SPEC(J), WTMOLE(J), BCFLAG(J)
  330 CONTINUE
C
C     ------  SCHEDULE AND INITIALIZE DIFFUSIVITIES
C
      IF(NHT.EQ.1)   GO TO 360
      K = NOSTAT + 3
      TSTOP = HTINV(NHT,1)
      CALL XMIT (NHT,HTINV(1,1),WORK(1))
      DO 350 J =2,K
      CALL SKEDUL(HTINV(1,J),WORK,NHT,NINT,UPDINT,INTIM,TSTOP,-1.,F,1)
      IF(F.EQ.YES)  GO TO 800
  350 CONTINUE
```

```
      NHT = NINT
      HTINV(1,1) = INTIM
      DO 355 I =2,NHT
      HTINV(I,1) = HTINV(I-1,1) + UPDINT
355   CONTINUE
      TSTOP = TSTOP2
360   CONTINUE
      K = NOSTAT + 1
      DO 370 I =1,K
      J = I + 2
      DFINIT(I) = HTINV(1,J)
370   CONTINUE
      CALL DIFCOF(NOSTAT)
C      ----     INITIALIZE RATEFF
      CALL XMIT(NOREAC,RATKON,RATEFF)
C
410   IPROD = NK*NOSTAT
      TDELZ(1) = 2./HTCELL(1)
      TDELZ(2) = 2./HTCELL(NOSTAT)
      CALL NEWPAG(TITLE,0,IDATE)
      WRITE(LOUT,14)
      HTCELL(1) = HTCELL(1)/2.
      HTCELL(NOSTAT) = HTCELL(NOSTAT)/2.
      I = 1
      WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)
      DO 420 I = 2,NOSTAT
      WRITE(LOUT,43) DFINIT(I), DELZ(I-1)
      IF(I.EQ.NOSTAT)  GO TO 415
      WRITE(LOUT,42)  I, ZEE(I), HTCELL(I)
      GO TO 420
415   WRITE(LOUT,44) I,ZEE(I),DFINIT(I),HTCELL(I)
420   CONTINUE
      HTCELL(1) = HTCELL(1)*2.
      HTCELL(NOSTAT) = HTCELL(NOSTAT)*2.
C
C     ----- INITIALIZE INTERNALLY COMPUTED CONCENTRATIONS
      CALL ISTATE
C     -----------  INITIALIZE YO VECTOR OF CONCENTRATIONS FOR DRIVE
      DO 425 I=1,NOSTAT
      K= NK*(I-1) + 1
      CALL XMIT(NK,CONIN(1,I),YO(K))
  425 CONTINUE
      CALL NEWPAG(TITLE,0,IDATE)
      ITIM = ITHOUR(TIME)
      WRITE(LOUT,97)   INTIM, ITIM
      WRITE(LOUT,18)   (SPEC(J),J=1,NOSPEC)
      DO 450 K=1,NOSTAT
      WRITE(LOUT,32)  ZEE(K), ( PPM(J,K),J=1,NOSPEC)
450   CONTINUE
C
      IF(SPUNCH.NE.YES)  GO TO 454
C     ------  WRITE I.C. ON TAPE LUP FOR PUNCH OR FILE STORAGE
      WRITE(LUP,80) TITLE
      WRITE(LUP,81) TIME,   (ZEE(K),K=1,NOSTAT)
      NOELV = 1
      DO 453 K=1,NOELV
      WRITE(LUP,81) ( PPM(J,K),J=1,NPUNCH)
453   CONTINUE
```

```
 454    CONTINUE
        IF(NCURV.EQ.0) GO TO 459
        KT = 1
        DO 455 I = 1,NCURV
        J = KPS(I)
        IF(J.NE.5)  GO TO 456
 C      SCALE CO FOR PLOTTING
        GRCON(I,KT) = .10*PPM(J,1)
        IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)
        GO TO 455
 456    CONTINUE
        GRCON(I,KT) = PPM(J,1)
        IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)
   455 CONTINUE
        TOUT(KT) = INTIM
   459 CONTINUE
 C      ----------   INITIALIZE ADDITIONAL PARAMETERS FOR INTEGRATION
        NSTEP = 0
        N = IPROD
        KNTER = 0
        TPRINT = INTIM + PRNTIN -.01
        RECALL = YES
        UPD4 = UPDINT/4.0
 C
 C      ------------- THE INTEGRATION TIME CYCLE BEGINS HERE -------------
 C
        CALL TIMEX(0,A,B)
   460 CONTINUE
        OLDTIM = TIME
        IF(RECALL .NE. YES)  GO TO 470
        INDEX = 1
        TOUTEP = OLDTIM + UPDINT/10.
        IF(KNTER .GT. 1)   DELT = .01
   470 IF(NSTEP .GT. NUMSTP)  GO TO 800
        TLAST = TIME
 C
 C                   INTEGRATE BY EPISODE
 C
        CALL DRIVE(N,TIME,DELT,YO,TOUTEP,EPS,IERROR,MF,INDEX,BIGSTP,KOK)
 C
 C
        IF(KOK.GE.0)  GO TO 475
        ISTOP = IES
        WRITE(LOUT,60)
        GO TO 480
 475    IF(INDEX .LT. 0)  GO TO 800
        INDEX = 2
        KNTER = KNTER + 1
        IF(RECALL.NE. YES)   GO TO 480
 C      MAKE SHORT CALL TO EPISODE (DRIVE) IF RECALL EQUALS YES
 C      THEN RESET TOUTEP TO ACTUAL TOUT ON NEXT CALL
        TOUTEP = OLDTIM + UPDINT
        RECALL = RNEG
        GO TO 470
 480    TOUTEP = OLDTIM + 2.*UPDINT
 C      ------------ UNPACK YO FOR CHECKING AND OUTPUT
        DO 500 I=1,NOSTAT
        K = NK*(I-1) + 1
```

```
      CALL XMIT(NK,YO(K),CONIN(1,I)  )
500   CONTINUE
620   CONTINUE
      CALL PRODUK(UPDINT)
C
      IF(TIME.LT.TPRINT)  GO TO 694
      TPRINT = TPRINT + PRNTIN
 650  CONTINUE
C
C     ------- OUTPUT
C
      CALL NEWPAG(TITLE,0,IDATE)
      IF(NOPFLX. EQ. 0)  GO TO 682
      DO 681 J=1,NDPFLX
      I= LOCDPF(J)
      SINK =- DPRATE(J)*(PPM(I,1)**DEPOWR(J))
  681 WRITE(10,88)  SPEC(I),  SINK
  682 CONTINUE
      ITIM = ITHOUR(TIME)
      WRITE(LOUT,91)  TIME, ITIM, RATEV(1,1), DFINIT(1)
      IF(NUMFLX.LE.0)  GO TO 656
      DO 655 I = 1,NUMFLX
      K = LOCFLX(I)
      WORK(I) = SPEC(K)
      WORK(I+10) = FLXWAL(K)*TDELZ(1)
655   CONTINUE
656   IF(NPSFLX.LE.0)  GO TO 661
      DO 660 I = 1,NPSFLX
      K = LOCPSF(I)
      WORK(I+20) = SPEC(K)
      WORK(I+30) = PSRATE(K,1)
660   CONTINUE
661   CONTINUE
      NMX = MAX0(NUMFLX,NPSFLX)
      NMV = NMX + 1
      NMX = NMX + 3
      DO 662 I = NMV,NMX
      IF(I.EQ.8) K=2
      IF(I.EQ.9) K=7
      IF(I.EQ.10) K=9
      WORK(I) = SPEC(K)
      WORK(I+10) = FLXWAL(K)*TDELZ(1)
      WORK(I+20) = SPEC(K)
      WORK(I+30) = PSRATE(K,1)
662   CONTINUE
      WRITE(LOUT,92)
      WRITE(LOUT,93)  (WORK(I),WORK(I+10)
     1 ,               WORK(I+20), WORK(I+30),I=1,NMX)
      WRITE(LOUT,94)
      WRITE(LOUT,9) (SPEC(J),J=1,NOSPEC)
      DO 690 K = 1,NOSTAT
      WRITE(LOUT,32) ZEE(K) , (PPM(J,K),J=1,NOSPEC)
  690 CONTINUE
      IF (TIMOUT.NE.YES)  GO TO 691
      WRITE(LOUT,66)  INDEX,HUSED,NQUSED,NSTEP,NFE,NJE
      NTSTP = NSTEP
      CALL TIMEX(NTSTP,A,B)
  691 CONTINUE
```

```
      IF(SPUNCH.NE.YES)  GO TO 694
C     ------  WRITE OUTPUT ON TAPE LUP FOR PUNCH OR FILE STORAGE
      WRITE(LUP,81) TIME
      DO 693 K=1,NOELV
      WRITE(LUP,81) (PPM(J,K),J=1,NPUNCH)
  693 CONTINUE
  694 CONTINUE
C
      IF(NCURV.EQ.0) GO TO 700
      KT = KT + 1
      IF(KT.GT.KTMAX) GO TO 700
      DO 695 I = 1,NCURV
      J = KPS(I)
      IF(J.NE.5)  GO TO 696
C     SCALE CO FOR PLOTTING
      GRCON(I,KT) = .10*PPM(J,1)
      IF(GRCON(I,KT).GT.VALMAX(I)) VALMAX(I)=GRCON(I,KT)
      GO TO 695
696   CONTINUE
      GRCON(I,KT) = PPM(J,1)
      IF(PPM(J,1).GT.VALMAX(I)) VALMAX(I) = PPM(J,1)
  695 CONTINUE
      TOUT(KT) = TIME
  700 CONTINUE
      IF(TIME.GE.(TSTOP-.01)) ISTOP = IES
      IF(ISTOP.EQ. IES)  GO TO 800
C
C     ------------ UPDATE EMISSION FLUXES
      IFLXTM = IFLXTM+1
      IF (IFLXTM.GT.NFLUX) IFLXTM = NFLUX
      CALL UPFLX1(TIME,IFLXTM)
      IPS = IPS + 1
      CALL UPSORC(TIME,IPS,NOSTAT,SPEC)
720   CCNTINUE
C
C     ------- UPDATE SKY COVER FACTOR
      IF (QRATE.EQ.RNEG) GO TO 730
      IF(QCLOUD.EQ.RNEG)  GO TO 725
      IF(ICLOUD.GE.NCLOUD)  GO TO 725
      BB= CLOUDT(ICLOUD+1)  - UPD4
      IF(TIME.LT. BB)  GO TO 725
      ICLOUC = ICLOUD + 1
      CLOUDY = CLOUDF(ICLOUD)
      WRITE(10,75)  TIME ,  CLOUDY
725   CONTINUE
C
C     -----------  UPDATE THE VARIABLE PHOTODISSOCIATION RATE CONSTANTS
C
      ROLOK1 = RATEV(1,1)
      CALL PHOTOR(TIME,NOSTAT,ZEE,CLOUDY,RATEV)
      IF(RATEV(1,1).LT..2*ROLOK1.OR.RATEV(1,1).GT.5.*ROLOK1)RECALL=YES
730   CONTINUE
C
C     ----           UPDATE DIFFUSION COEFFICIENTS
C
      IF(NHT.EQ.1)  GO TO 770
      AA= HTINV(IHT,1) - UPD4
      IF(TIME.LT.AA)  GO TO 770
      IF(KHT.GE.NHT)  GO TO 770
      IHT = IHT + 1                                                     NKM 1036
      KHT = KHT + 1                                                     NKM 1037
      WRITE(11,46) TIME                                                 NKM 1038
      K = NOSTAT+1                                                      NKM 1039
      DO 750 I = 1,K                                                    NKM 1040
      DFINIT(I) = HTINV(IHT,I+2)                                        NKM 1041
  750 CONTINUE                                                          NKM 1042
      WRITE(11,96) (DFINIT(I),I=1,K)                                    NKM 1043
      CALL DIFCCF(NOSTAT)                                               NKM 1044
      IF (IHT.GT.NHT) IHT = NHT                                         NKM 1045
  770 CONTINUE                                                          NKM 1046
C                                                                       NKM 1047
C       -----  UPDATE TEMPERATURE DEPENDENT REACTION RATES              NKM 1048
      IF(VTEMP.NE.YES)  GO TO 780                                       NKM 1049
      IF(ITEMP.GE.NTEMP)  GO TO 780                                     NKM 1050
      IP1 = ITEMP + 1                                                   NKM 1051
      BB = TMTEMP(IP1) - UPD4                                           NKM 1052
      IF(TIME.LT.BB)  GO TO 780                                         NKM 1053
      ITEMP = IP1                                                       NKM 1054
      CALL TEMPR(ITEMP,TIME,TEMP)                                       NKM 1055
  780 CONTINUE                                                          NKM 1056
C                                                                       NKM 1057
      GO TO 460                                                         NKM 1058
C                                                                       NKM 1059
C     ------------  EXIT FROM INTEGRATION TIME CYCLE ----------------   NKM 1060
C                                                                       NKM 1061
  800 CONTINUE                                                          NKM 1062
      IF(F.EQ.YES)  RETURN                                              NKM 1063
      IF(INDEX.LT.0)  WRITE(LOUT,74) TIME, INDEX                        NKM 1064
      IF(NCURV.EQ.0) GO TO 805                                          NKM 1065
      IF(KT.GT.KTMAX) KT = KTMAX                                        NKM 1066
      CALL COPLOT(KT)                                                   NKM 1067
  805 CONTINUE                                                          NKM 1068
      IF(NSTEP .GT. NUMSTP)  WRITE(LOUT,26)                             NKM 1069
      RETURN                                                            NKM 1070
C                                                                       NKM 1071
C     FORMATS                                                           NKM 1072
C                                                                       NKM 1073
    8 FORMAT (1H ,5X,A4,8H FLUX = ,2X,G15.8)                            NKM 1074
    9 FORMAT (1H0,4X,9HHEIGHT-M.,3X,A4,7(11X,A4)/,6X,8(11X,A4),/,       NKM 1075
     1  6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/,6X,8(11X,A4),/     NKM 1076
   13 FORMAT (1H0,  /,  34X,48HCONCENTRATION PROFILES-PARTS PER MILLION ) NKM 1077
     1        //45X,5HTIME=,E12.5,8H MINUTES//10X,20HGROUND DIFFUSIVITY NKM 1078
     2=,G15.8,10X, 1HK,I1,3H = ,G12.4,10X,1HK,I2,3H = ,G12.4)           NKM 1079
   14 FORMAT (1H0,10X,30HINITIAL DIFFUSION COEFFICIENTS//10X,52HVALUES SNKM 1080
     1HOWN ARE FOR POINTS HALFWAY BETWEEN STATIONS///10X,7HSTATION,10X,1NKM 1081
     25HHEIGHT (METERS),10X,11HDIFFUSIVITY,16X,7HDELTA Z,11X,11HCELL HEINKM 1082
     3GHT  )                                                            NKM 1083
   15 FORMAT (1H1,10X,84HNUMBER OF NO2 RATE CONSTANTS EXCEEDED THE DIMENNKM 1084
     1SIONS OF VECTOR -RATK1-, JOB ABORTED.)                            NKM 1085
   16 FORMAT (1H0,//,55H  INPUT DATA FOR VERTICAL DIFFUSION WITH PHOTOCHNKM 1086
     1EMISTRY  ///8X,15HINITIAL TIME IS,F25.2,8H MINUTES//8X,           NKM 1087
     1   13HFINAL TIME IS        ,                                      NKM 1088
     1       F27.2,8H MINUTES//8X,25HINITIAL TIME STEP SIZE IS,F15.6,8H MNKM 1089
     3INUTES   //,8X,25HMAXIMUM TIME STEP SIZE IS ,F15.2,               NKM 1090
     3 8H MINUTES  ,//,8X,22HNUMBER OF REACTIONS IS ,I18,               NKM 1091
     4    //8X,20HNUMBER OF SPECIES IS ,I20,//,8X, 35HNUMBER OF SPECIES NKM 1092
```

```
      5IN STEADY STATE IS,I4,
      5               //8X,33HNUMBER OF VERTICAL MESH POINTS IS,I7,4X,33HNKM 1094
      7INCLUDING THE GROUND AND THE EDGE )                              NKM 1095
 17    FORMAT (1H0,//13X,8(8X,A4),6X,13HRATE CONSTANT,/,13X,8(8X,A4),   NKM 1096
      1 /,13X,8(8X,A4),/,13X,8(8X,A4),/,13X,8(8X,A4) )                  NKM 1097
 18    FORMAT(1H0,                                                      NKM 1098
      1/////4X,9HHEIGHT-M.,5X,A4,7(11X,A4)/,7X,8(11X,A4),/,7X,8(11X,A4),NKM 1099
      2  /,7X,8(11X,A4),/,7X,8(11X,A4)   )                              NKM 1100
 20    FORMAT(1H0,5X,I3,2X,8F12.2,10X,G12.5)                            NKM 1101
 21    FORMAT (1H ,10X,8F12.2,/, 3(11X,8F12.2,/)  )                     NKM 1102
 22    FCRMAT(50H1  THE READ PHOTOLYTIC RATES OPTION WAS DELETED        NKM 1103
 26    FORMAT (1H1,25(1H /),10X,36HNUMBER OF INTEGRATION STEPS EXCEEDED,2NKM 1104
      14H LIMIT.  JOB TERMINATED.)                                      NKM 1105
 27    FORMAT(1H ,I3,3H.  ,A4,14X,F7.1,16X,I2)                          NKM 1106
 28    FORMAT(1H0,/6X,7HSPECIES,7X,16HMOLECULAR WEIGHT,5X,9HBOUNDARY    NKM 1107
      1  11HCONDITION    ,/)                                            NKM 1108
 29    FORMAT (40X,A4)                                                  NKM 1109
 30    FORMAT (40X,F10.0)                                               NKM 1110
 31    FORMAT (40X,I10)                                                 NKM 1111
 32    FORMAT (1H ,2X,F7.2,8E15.5,/, 4(10X,8E15.5,/) )                  NKM 1112
 33    FORMAT (1H1,25(/),25X,20A4//25X,8HDATE -- ,A10,12X,10HCP TIME = ,FNKM 1113
      110.2,54  SEC)                                                    NKM 1114
 37    FORMAT(1H0,    20X,17HR E A C T I O N S ,55X,30HRATE CONSTANTS (1/NKM 1115
      1(PPM--MIN))  /)                                                  NKM 1116
 38    FORMAT(1H ,3X,I3,2H.  ,20A4,8X,E12.4)                            NKM 1117
 39    FORMAT (11X,A4,5X,11HAT STATION ,I3,5X,25HNEGATIVE OR ZERO AT TIMENKM 1118
      1 ,E10.3,5X,18HCONCENTRATION IS  ,E10.3,5X,8HDELT IS ,E10.3)      NKM 1119
 40    FORMAT (20A4)                                                    NKM 1120
 41    FORMAT (20X,5E10.4)                                              NKM 1121
 42    FORMAT(1H0,10X,I5,15X,F10.2,55X,F10.2)                           NKM 1122
 43    FORMAT(1H0,50X,G15.8,10X,F10.2)                                  NKM 1123
 44    FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8,30X,F10.2)                 NKM 1124
 45    FORMAT (1H1,10X,57HTOO MANY INVERSION HEIGHT ENTRIES IN TABLE.  JONKM 1125
      1B ABORTED.)                                                      NKM 1126
 46    FORMAT(13H KZ AT TIME    ,F10.2)                                 NKM 1127
 47    FORMAT(1H0,10X,I5,15X,F10.2,10X,G15.8)                           NKM 1128
 51    FORMAT (40X,A4,6X,F10.2,I10)                                     NKM 1129
 52    FORMAT(20X,5(I2,8X)/20X,5(I2,8X))                                NKM 1130
 53    FORMAT (40X,I10,2F10.0)                                          NKM 1131
 54    FORMAT (20X,5E12.4)                                              NKM 1132
 55    FORMAT (20X,10F6.2)                                              NKM 1133
 57    FORMAT (1H1,10X,45HTOO MANY CLOUD COVER ENTRIES  --  JOB ABORTED) NKM 1134
 58    FORMAT(10(/),10X,73HDEPLETION OF  NO  HAS CAUSED FAILURE OF CHEMICNKM 1135
      1AL MODEL.  CASE TERMINATED. )                                    NKM 1136
 59    FORMAT (40X,2F10.0)                                              NKM 1137
 60    FORMAT(1H1,/////,20X,41HJOB TERMINATED -- NEGATIVE CONCENTRATIONS NKM 1138
      1    30H OR DEPLETION OF NITRIC OXIDE      )                      NKM 1139
 66    FORMAT(1H0,9X,20HINTEGRATION INDEX = ,I4,10X,18HLAST STEP SIZE =  NKM 1140
      1    E10.3,20X,17HLAST ORDER USED = ,I3,/,10X,18HNUMBER OF STEPS = ,NKM 1141
      2 I6,10X,33HNUMBER OF FUNCTION EVALUATIONS = ,I4,10X,34H NUMBER OF NKM 1142
      3JACOBIAN EVALUATIONS = ,I4)                                      NKM 1143
 68    FORMAT(1H0, 5X,70HREVISED AREA SOURCE FLUX SCHEDULE       ---    NKM 1144
      1     PPM-METERS/MINUTE   ,//,6X,4HTIME,7(11X,A4),/,10X, 3(11X,A4)) NKM 1145
 69    FORMAT(1H ,F10.1,1X,7E15.4)                                      NKM 1146
 74    FORMAT (1H0,//,8H  TIME =, F8.2,5X,9HINDEX =    ,I2,             NKM 1147
      1  36H  INTEGRATION TERMINATED BY EPISODE   )                     NKM 1148
 75    FORMAT (1H0,10X,33HCLEARNESS RATIO UPDATED AT TIME =   ,F8.2,5X, NKM 1149
      1     20HSKY COVER FACTOR  =    , F8.4)                           NKM 1150
                                                                        NKM 1151
```

```
76    FORMAT(1H1,9X,41HJOB ABORTED BECAUSE OF TOO MANY SPECIES  , //
     1/,9H NOSPEC =  ,I3,5X,9H  NSTDY=   ,I3,5X,5H NK=   ,I3)
77    FORMAT(1H1,58HJOB ABORTED BECAUSE OF TOO FEW VERTICAL STATIONS  ,/
     1//,10X,8HNOSTAT = , I2)
79    FORMAT (47H0  TOO MANY TEMPERATURES INPUT -- JOB ABORTED       )
80    FORMAT (20A4)
81    FORMAT ( 7(6E12.4,/))
83    FORMAT (1H0,5X,50HPOINT SOURCE EMISSION RATES BY VERTICAL CELL
     1     10X,10H(PPM/MIN)  ,//,6X,4HTIME,2X,8(8X,A4))
84    FORMAT (1H0,F9.1,4X,8E12.3)
85    FORMAT (1H ,13X,8E12.3)
88    FORMAT(1H ,5X,A4,8H SINK = ,2X,G15.8)
91    FORMAT(1H0,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,
     1 17H ON 2400-CLOCK)   /,40X,24HSURFACE CELL PARAMETERS
     2/ 40X,30HNO2 PHOTODISSOCIATION RATE  =   ,F8.3,10H   (/MIN)  ,/,
     3 40X,30HDIFFUSIVITY COEFFICIENT     =   ,F8.1,13H   (M**2/MIN))
92    FORMAT(1H ,39X,12HAREA SOURCES,2X,9H(PPM/MIN)3X,13HPOINT SOURCES
     1  2X,9H(PPM/MIN) )
93    FORMAT(  10(/,43X,A4,3H = ,E11.3,8X,A4,3H = ,E11.3))
94    FCRMAT(       40X,43HCONCENTRATION PROFILES IN PARTS PER MILLION )
96    FORMAT(1H ,10X,6F15.2)
97    FORMAT(1H0,/,40X,6HTIME =,F8.2,14H MINUTES     ( ,  I4,
     1 17H ON 2400-CLOCK)   //,
     1/,40X,51HINITIAL CONCENTRATION PROFILES IN PARTS PER MILLION/)
99    FORMAT(11(//),57X,24HTHE ERT TRAJECTORY MODEL ,/57X,6(4H----),
     1  //,59X,20H(CODE DATE   2.1.79) ////,63X,12HDEVELOPED BY  ,//
     1  40X,41HENVIRONMENTAL RESEARCH   TECHNOLOGY, INC.  //
     1  58X,22HSANTA BARBARA DIVISION   )
      END


      SUBROUTINE PHOTOR (TIME,NOSTAT,ZEE,CLOUDY,RATEV)
C
C        PHOTOR CALCULATES PHOTODISSOCIATION RATES FOR THE  REACTIONS
C
C      1.     NO2 + HV   ----  NO + O
C      2.     HONO + HV  ----  NO + OH
C      3.     RCHO + HV  ----  RO2 + HO2 + CO
C      4.     HCHO + HV  ----- HO2 + CO
C
C ---  INPUTS ----
C    TIME = TIME IN MINUTES FROM MIDNIGHT
C    NOSTAT = NUMBER OF VERTICAL MESH POINTS
C    ZEE = ELEVATION OF VERTICAL MESH POINTS ABOVE SURFACE (METERS)
C    CLOUDY = SKY CLEARNESS RATIO
C
C --- OUTPUTS ---
C    RATEV = 2-D ARRAY OF RATE CONSTANTS
C
      DIMENSION RATEV(4,1),ZEE(1)
      DIMENSION ZINP(10),HTINP(8),R(8,10,4)
      DIMENSION RNO2(8,10),RHONO(8,10), RRCHO(8,10), RHCHO(8,10)
      DIMENSION RMIN(4), REXP(4), NAM(4)
C
      COMMON /CHEM4/  RLAT, RLONG, TMZONE, HIRATE, JDATE
C
      EQUIVALENCE( RNO2(1,1),R(1,1,1))
      EQUIVALENCE(RHONO(1,1),R(1,1,2))
      EQUIVALENCE(RRCHO(1,1),R(1,1,3))
      EQUIVALENCE(RHCHO(1,1),R(1,1,4))
C
      DATA RMIN /1.E-3, 1.E-4,1.E-6,1.E-5 /
      DATA REXP /1.0, 0.100, 0.0060, 0.0100/
      DATA NHTI,NZINP,NR /8,10,4/
      DATA YES, NAM /3HYES,3HNO2,4HHONO,4HRCHO,4HHCHO/
      DATA ZINP /0.,10.,20.,30.,40.,50.,60.,70.,78.,86. /
      DATA HTINP /0.,150.,360.,640.,980.,1380.,1840.,4210./
C
      DATA RNO2 /
     1  .534,.565,.594,.621..648,.671..691,.758,
     2 .529,.561,.589,.617,.644,.668,.688,.755,
     3 .516,.549,.579,.607,.633,.657,.678,.746,
     4 .493,.526,.557,.586,.614,.638..660,.720,
     5 .457,.492,.523,.553..582,.607,.629,.703,
     5 .404,.439,.472,.503,.531,.556,.579,.657,
     7 .325,.360,.392,.423,.451,.476,.499,.581,
     8 .214,.243,.272,.300,.326,.348,.369,.451,
     9 .105,.123,.141..160,.179,.196,.211,.280,
     1 .023,.025,.028,.031,.034,.037,.040,.050/
      DATA  RHONO/
     1.958,1.02,1.08,1.13,1.18,1.23,1.27,1.40,
     2.950,1.01,1.07,1.12,1.17,1.22,1.26,1.40,
     3.925,.987,1.05,1.10,1.15,1.20,1.24,1.38,
     4.881,.945,1.00,1.06,1.12,1.16,1.21,1.35,
     5.812,.877,.938,.996,1.05,1.10,1.45,1.29,
     6.710,.777,.837,.895,.951,1.00,1.04,1.20,
     7.563,.626,.685,.741,.795,.843,.886,1.05,
     8.361,.411,.460,.508,.554,.596,.634,.794,
     9.174,.200,.229,.260,.289,.317,.344,.468,
     1.039,.043,.048,.052,.057,.061,.656,.878/
      DATA  RRCHO /
     1.230,.249,.267,.286,.306,.324,.341,.397,
     2.227,.246,.264,.283,.301,.319,.367,.393,
     3.216,.234,.252,.270,.289,.307,.323,.379,
     4.196,.214,.231,.249,.267,.284,.300,.355,
     5.169,.186,.202,.218,.235,.251,.266,.320,
     6.134,.147,.162,.177,.191,.206,.219,.269,
     7.092,.103,.114,.125,.136,.147,.159,.202,
     8.047,.054,.060,.067,.074,.081,.088,.118,
     9.019,.021,.023,.026,.029,.032,.035,.050,
     1.003,.004,.004,.004,.005,.005,.005,.006/
      DATA  RHCHO /
     X .614,.662,.709,.756,.802,.846,.886,1.02,
     1 .606,.654,.700,.747,.794,.838,.878,1.01,
     2 .581,.629,.675,.721,.767,.810,.850,.985,
     3 .538,.584,.629,.675,.719,.762,.801,.937,
     4 .474,.519,.562,.605,.648,.689,.726,.861,
     5 .388,.428,.469,.508,.547,.585,.620,.750,
     6 .280,.313,.346,.379,.412,.444,.474,.594,
     7 .156,.178,.199,.222,.244,.266,.288,.381,
     7 .067,.076,.086,.096,.107,.117,.128,.181,
     8 .013,.015,.016,.018,.019,.021,.022,.029/
C
      STIME = ITHOUR(TIME)
      IY = JDATE/10000
      IM = JDATE/100 - IY*100
      ID = JDATE - IY*10000 - IM*100
      IY = IY + 1900
C    RETRIEVE SOLAR ZENITH ANGLE ZEN
      CALL SOLAR(RLAT,RLONG,TMZONE,IY,IM,ID,STIME,D,5)
      ZEN = 90. - D
      IF((ZEN.GE.ZINP(1)) .AND. (ZEN.LE.ZINP(NZINP))) GO TO 40
C
      DO 20 K = 1,NOSTAT
      DO 20 J = 1,NR
      RATEV(J,K) = RMIN(J)
20    CONTINUE
      GO TO 200
C
40    DO 60 I = 1,NZINP
      IF(ZINP(I).LT.ZEN)  GO TO 60
      I1  = I - 1
      I2 = I
      Z1 = ZINP(I1)
      Z2 = ZINP(I2)
      GO TO 80
60    CONTINUE
80    PZ = (ZEN-Z1)/(Z2-Z1)
      J = 1
      DO 90 KJ = 1,NHTI
      IF(ZEE(JJ+1).GT.HTINP(J+1))  J = J + 1
90    CONTINUE
C
      DO 140 JJ = 1,NOSTAT
      JP = J + 1
      PH = (ZEE(JJ)-HTINP(J))/(HTINP(JP)-HTINP(J))
      DO 100 K = 1,NR
      R1 = R(J,I1,K) +PZ*(R(J,I2,K)-R(J,I1,K))
      R2 = R(JP,I1,K) + PZ*(R(JP,I2,K)-R(JP,I1,K))
C
      RATEV(K,JJ) = (R1 + PH*(R2-R1))*REXP(K)*CLOUDY
C
100   CONTINUE
      IF(HIRATE.NE.YES)  GO TO 160
      IF(JJ.EQ.NOSTAT)  GO TO 140
      KK = J
      DO 120 KJ = KK,NHTI
      IF(ZEE(JJ+1).GT.HTINP(J+1))  J = J + 1
120   CONTINUE
      J = MIN0(J,NHTI)
140   CONTINUE
      GO TO 200
160   DO 180 K = 2,NOSTAT
      DO 180 J = 1,NR
      RATEV(J,K) = RATEV(J,1)
180   CONTINUE
200   WRITE(10,2)  TIME,ZEN
      DO 220 J= 1,NR
      WRITE  (10,1)       NAM(J),(RATEV(J,K),K=1,NOSTAT)
220   CONTINUE
      RETURN
1     FORMAT(1H ,A4,8E10.3)
2     FORMAT( 30H  PHOTOLYSIS RATES AT TIME =  ,F7.1,5X,
     1  15HZENITH ANGLE =      ,F7.2)
      END


      SUBROUTINE PRODUK(DT)
C
C          COMPUTES APPROXIMATE PRODUCT SPECIES CONCENTRATIONS
C
C          HNO3  OZID  H2O2  NTRA  ANO2  ACHO
C
C          THIS ROUTINE IS SPECIFIC TO THE ERT PHOTOCHEMICAL MECHANISM
C     39 SPECIES  X  64 REACTIONS     11.1.78
      COMMON/CHEM1/ NOSTAT,                NOSTM1,                NOREAC,
     1              NOSPEC,                NSTDY,                 NK
      COMMON/CHEM2/ CONIN(40,5),           WTMOLE(40),            RATKON(65),
     1              RATEFF(65),            RATEV(4,5),            QRATE,
     2              NVRATE,                LOCVRT(4)
      DO 10 K = 1,NOSTAT
C          ----------  HNO3   ---------
      RATE = RATEFF(8)*CONIN(2,K)*CONIN(31,K)  +  RATEFF(16)*CONIN(21,K)
     1 *CONIN(33,K)
      CONIN(34,K) = CONIN(34,K) + DT*RATE
C          ----------  OZID  -----------
      RATE = RATEFF(31)*CONIN(6,K)*CONIN(25,K) + RATEFF(32)*CONIN(7,K)*
     1  CONIN(25,K) + RATEFF(33)*CONIN(6,K)*CONIN(26,K) + RATEFF(34)*
     2  CONIN(7,K)*CONIN(26,K)
      CONIN(35,K) = CONIN(35,K) + RATE*DT
C          ----------  H2O2  -----------
      RATE = (CONIN(22,K)**2)*RATEFF(12) + CONIN(22,K)*CONIN(13,K)*
     1                                                        RATEFF(55)
      CONIN(36,K) = CONIN(36,K) + RATE*DT
C          ----------  NTRA  -----------
      RATE = CONIN(1,K)*CONIN(28,K)*RATEFF(37) + CONIN(2,K)*CONIN(27,K)
     1                                              *RATEFF(41)*.85
      CONIN(37,K) = CONIN(37,K) + RATE*DT
C          ----------  ANO2  -----------
      RATE = CONIN(2,K)*CONIN(13,K)*RATEFF(54)
      CONIN(38,K) = CONIN(38,K) + RATE*DT
C          ----------  ACHO  -----------
      RATE = CONIN(1,K)*CONIN(16,K)*RATEFF(58)
      CONIN(39,K) = CONIN(39,K) + RATE*DT
   10 CONTINUE
      RETURN
      END


      SUBROUTINE RATEHI(K)
C
C       RETRIEVES PHOTODISSOCIATION RATES FOR THE KTH MESH POINT
C
      COMMON /CHEM2/ CONIN(40,5),  WTMOLE(40),  RATKON(65),
     1               RATEFF(65),  RATEV(4,5),  QRATE, NVRATE, LOCVRT(4)
C
      DO 10 I = 1,NVRATE
      J = LOCVRT(I)
      RATKON(J) = RATEV(I,K)
      RATEFF(J) = RATEV(I,K)
   10 CONTINUE                                                          NKM 1375
      RETURN                                                            NKM 1376
      END                                                               NKM 1377
                                                                        NKM 1378



      SUBROUTINE RATES(Y,YDOT)
C                                                                       NKM 1379
C          CALCULATION OF CHEMICAL RATES                                NKM 1380
C                                                                       NKM 1381
      INTEGER BCFLAG                                                    NKM 1382
      COMMON/CHEM1/ NOSTAT,                NOSTM1,             NOREAC,  NKM 1383
     1              NOSPEC,                NSTDY,              NK       NKM 1384
      COMMON/CHEM2/ CONIN(40,5),           WTMOLE(40),                  NKM 1385
     1              RATEFF(65),            RATEV(4,5),         RATKON(65), NKM 1386
     2              NVRATE,                LOCVRT(4)           QRATE,   NKM 1387
      DIMENSION Y(1), YDOT(1)   ,  RATE(40),  C(40),  R(65)             NKM 1388
      EQUIVALENCE (R,RATEFF)                                            NKM 1389
      DATA  YES  /3HYES/                                                NKM 1390
      NKP1 = NK + 1                                                     NKM 1391
      DO 130 K = 1,NOSTAT                                               NKM 1392
      KS = NK*(K-1)                                                     NKM 1393
      DO 90  I=1,NK                                                     NKM 1394
      C(I) = Y(I+KS)                                                    NKM 1395
   90 CONTINUE                                                          NKM 1396
      IF(NSTDY.EQ.0)  GO TO 105                                         NKM 1397
      DO 100 I=NKP1,NOSPEC                                              NKM 1398
      C(I) = CONIN(I,K)                                                 NKM 1399
  100 CONTINUE                                                          NKM 1400
  105 CONTINUE                                                          NKM 1401
C                                                                       NKM 1402
      IF(QRATE.EQ.YES)  CALL RATEHI(K)                                  NKM 1403
      CALL UNMIXR(K)                                                    NKM 1404
C                                                                       NKM 1405
C      EXPLICIT CHEMICAL RATE EQUATIONS                                 NKM 1406
C                                                                       NKM 1407
C          RATES FOR THE ERT PHOTOCHEMICAL MECHANISM  (39X64)  1.31.79  NKM 1408
C                                                                       NKM 1409
      RATE(  1) = + R(  1)*C(  2) - R(  3)*C(  1)*C(  3) - R(  4)*C(  1)NKM 1410
     *    *C(  2)*C( 33) + R(  5)*C(  4)** 2 + R(  6)*C(  4) - R(  7)   NKM 1411
     *    *C(  1)*C( 31) - R( 10)*C(  1)*C( 22) - R( 14)*C(  1)*C( 19)  NKM 1412
     *     - R( 20)*C(  1)*C( 18) - R( 27)*C(  1)*C( 25) - R( 28)*C(  1)NKM 1413
     *    *C( 26) - R( 36)*C(  1)*C( 28) - R( 37)*C(  1)*C( 28) - R( 39)NKM 1414
     *    *C(  1)*C( 23) - R( 44)*C(  1)*C( 24) - R( 53)*C(  1)*C( 13)  NKM 1415
     *     - R( 58)*C(  1)*C( 16)                                       NKM 1416
      RATE(  2) = - R(  1)*C(  2) + R(  3)*C(  1)*C(  3) - R(  4)*C(  1)NKM 1417
     *    *C(  2)*C( 33) + R(  5)*C(  4)** 2 - R(  8)*C(  2)*C( 31) +   NKM 1418
     *    R( 10)*C(  1)*C( 22) - R( 11)*C(  2)*C( 22) - R( 13)*C(  2)   NKM 1419
     *    *C(  3) +  2.00*R( 14)*C(  1)*C( 19) - R( 15)*C(  2)*C( 19) + NKM 1420
     *    R( 17)*C( 21) + R( 20)*C(  1)*C( 18) + R( 22)*C( 20) + R( 27) NKM 1421
     *    *C(  1)*C( 25) + R( 28)*C(  1)*C( 26) - R( 29)*C(  2)*C( 25)  NKM 1422
     *     - R( 30)*C(  2)*C( 26) + R( 36)*C(  1)*C( 28) + R( 39)*C(  1)NKM 1423
     *    *C( 23) - R( 41)*C(  2)*C( 27) + R( 44)*C(  1)*C( 24) - R( 45)NKM 1424
     *    *C(  2)*C( 24) + R( 46)*C( 11) + R( 53)*C(  1)*C( 13) - R( 54)NKM 1425
     *    *C(  2)*C( 13) + R( 58)*C(  1)*C( 16)                         NKM 1426
      RATE(  3) = + R(  2)*C( 32) - R(  3)*C(  1)*C(  3) - R( 13)*C(  2)NKM 1427
     *    *C(  3) - R( 25)*C(  3)*C(  8) - R( 26)*C(  3)*C(  9)         NKM 1428
                                                                        NKM 1429
```

```
 RATE(  4) = +  2.00*R(  4)*C(  1)*C(  2)*C( 33) -  2.00*R(  5)  NKM 1430
*    *C(  4)** 2 - R(  6)*C(  4) + R(  7)*C(  1)*C( 31) +        NKM 1431
*     0.15*R( 41)*C(  2)*C( 27)                                  NKM 1432
 RATE(  5) = - R(  9)*C(  5)*C( 31) +  0.40*R( 24)*C(  9)*C( 32) + NKM 1433
*    R( 43)*C(  7) + R( 47)*C(  6) + R( 48)*C(  6)*C( 31)        NKM 1434
 RATE(  6) = +  1.50*R( 21)*C( 17) +  0.50*R( 25)*C(  3)*C(  8) + NKM 1435
*    R( 26)*C(  3)*C(  9) + R( 27)*C(  1)*C( 25) + R( 29)*C(  2)  NKM 1436
*    *C( 25) - R( 31)*C(  6)*C( 25) - R( 33)*C(  6)*C( 26) +     NKM 1437
*     0.50*R( 38)*C( 27) - R( 47)*C(  6) - R( 48)*C(  6)*C( 31) + NKM 1438
*    R( 61)*C( 25)*C( 29)                                        NKM 1439
 RATE(  7) = +  0.50*R( 21)*C( 17) +  0.30*R( 23)*C(  8)*C( 32) + NKM 1440
*     0.30*R( 24)*C(  9)*C( 32) +  0.50*R( 25)*C(  3)*C(  8) +   NKM 1441
*    R( 28)*C(  1)*C( 26) + R( 30)*C(  2)*C( 26) - R( 32)*C(  7)  NKM 1442
*    *C( 25) - R( 34)*C(  7)*C( 26) +  0.50*R( 38)*C( 27) + R( 40) NKM 1443
*    *C( 27) +  0.15*R( 41)*C(  2)*C( 27) - R( 42)*C(  7)*C( 31) - NKM 1444
*    R( 43)*C(  7) + R( 62)*C( 26)*C( 29)                        NKM 1445
 RATE(  8) = - R( 18)*C(  8)*C( 31) - R( 23)*C(  8)*C( 32) - R( 25)NKM 1446
*    *C(  3)*C(  8)                                              NKM 1447
 RATE(  9) = - R( 19)*C(  9)*C( 31) - R( 24)*C(  9)*C( 32) - R( 26)NKM 1448
*    *C(  3)*C(  9)                                              NKM 1449
 RATE( 10) = - R( 35)*C( 10)*C( 31)                              NKM 1450
 RATE( 11) = + R( 45)*C(  2)*C( 24) - R( 46)*C( 11)              NKM 1451
 RATE( 12) = - R( 50)*C( 12)*C( 31) - R( 51)*C( 12)*C( 31) - R( 57)NKM 1452
*    *C( 12)*C( 31)                                              NKM 1453
 RATE( 13) = + R( 51)*C( 12)*C( 31) - R( 53)*C(  1)*C( 13) - R( 54)NKM 1454
*    *C(  2)*C( 13) - R( 55)*C( 13)*C( 22)                       NKM 1455
 RATE( 14) = + R( 50)*C( 12)*C( 31) - R( 52)*C( 14)*C( 31) + R( 55)NKM 1456
*    *C( 13)*C( 22)                                              NKM 1457
 RATE( 15) = + R( 53)*C(  1)*C( 13) - R( 56)*C( 15)              NKM 1458
 RATE( 16) = + R( 57)*C( 12)*C( 31) - R( 58)*C(  1)*C( 16)       NKM 1459
 RATE( 17) = + R( 20)*C(  1)*C( 18) - R( 21)*C( 17) + R( 64)*C( 18)NKM 1460
*    *C( 29)                                                     NKM 1461
 RATE( 18) = + R( 18)*C(  8)*C( 31) + R( 19)*C(  9)*C( 31) - R( 20)NKM 1462
*    *C(  1)*C( 18) - R( 64)*C( 18)*C( 29)                       NKM 1463
 RATE( 19) = + R( 13)*C(  2)*C(  3) - R( 14)*C(  1)*C( 19) - R( 15)NKM 1464
*    *C(  2)*C( 19) + R( 17)*C( 21) + R( 29)*C(  2)*C( 25) + R( 30)NKM 1465
*    *C(  2)*C( 26)                                              NKM 1466
 RATE( 20) = + R( 11)*C(  2)*C( 22) - R( 22)*C( 20)              NKM 1467
 RATE( 21) = + R( 15)*C(  2)*C( 19) - R( 16)*C( 21)*C( 33) - R( 17)NKM 1468
*    *C( 21)                                                     NKM 1469
 RATE( 22) = + R(  9)*C(  5)*C( 31) - R( 10)*C(  1)*C( 22) - R( 11)NKM 1470
*    *C(  2)*C( 22) -  2.00*R( 12)*C( 22)** 2 + R( 21)*C( 17) +  NKM 1471
*    R( 22)*C( 20) +  0.40*R( 23)*C(  8)*C( 32) +  0.40*R( 24)   NKM 1472
*    *C(  9)*C( 32) +  0.40*R( 26)*C(  3)*C(  9) + R( 40)*C( 27) + NKM 1473
*    R( 43)*C(  7) +        R( 47)*C(  6) + R( 48)*C(  6)*C( 31) -  NKM 1474
*    R( 49)*C( 22)*C( 23) + R( 50)*C( 12)*C( 31) + R( 52)*C( 14)  NKM 1475
*    *C( 31) - R( 55)*C( 13)*C( 22) + R( 58)*C(  1)*C( 16) - R( 60)NKM 1476
*    *C( 22)*C( 29)                                              NKM 1477
 RATE( 23) = +  0.40*R( 23)*C(  8)*C( 32) +  0.40*R( 24)*C(  9)   NKM 1478
*    *C( 32) +  0.10*R( 25)*C(  3)*C(  8) +  0.15*R( 36)*C(  1)   NKM 1479
*    *C( 28) + R( 38)*C( 27) - R( 39)*C(  1)*C( 23) + R( 43)*C(  7)NKM 1480
*     + R( 44)*C(  1)*C( 24) - R( 49)*C( 22)*C( 23) + R( 56)*C( 15)NKM 1481
*     - R( 63)*C( 23)*C( 29)                                     NKM 1482
 RATE( 24) = + R( 42)*C(  7)*C( 31) - R( 44)*C(  1)*C( 24) - R( 45)NKM 1483
*    *C(  2)*C( 24) + R( 46)*C( 11)                              NKM 1484
 RATE( 25) = +  0.40*R( 25)*C(  3)*C(  8) +  0.80*R( 26)*C(  3)   NKM 1485
*    *C(  9) - R( 27)*C(  1)*C( 25) - R( 29)*C(  2)*C( 25) - R( 31)NKM 1486
*    *C(  6)*C( 25) - R( 32)*C(  7)*C( 25) - R( 61)*C( 25)*C( 29)  NKM 1487
```

```
      RATE( 26) = +  0.40*R( 25)*C(  3)*C(  8) - R( 28)*C(  1)*C( 26) -
     *    R( 30)*C(  2)*C( 26) - R( 33)*C(  6)*C( 26) - R( 34)*C(  7)
     *    *C( 26) - R( 62)*C( 26)*C( 29)
      RATE( 27) = +  0.85*R( 36)*C(  1)*C( 28) - R( 38)*C( 27) + R( 39)
     *    *C(  1)*C( 23) - R( 40)*C( 27) - R( 41)*C(  2)*C( 27) + R( 63)
     *    *C( 23)*C( 29)
      RATE( 28) = + R( 35)*C( 10)*C( 31) - R( 36)*C(  1)*C( 28) - R( 37)
     *    *C(  1)*C( 28)
C
      IF(NK.LE.28)  GO TO 110
C
C         RATES FOR SO2 AND SO4
C
      RATE( 29) = - R( 59)*C( 29)*C( 31) - R( 60)*C( 22)*C( 29) - R( 61)
     *    *C( 25)*C( 29) - R( 62)*C( 26)*C( 29) - R( 63)*C( 23)*C( 29)
     *     - R( 64)*C( 18)*C( 29)
      RATE( 30) = + R( 59)*C( 29)*C( 31) + R( 60)*C( 22)*C( 29) + R( 61)
     *    *C( 25)*C( 29) + R( 62)*C( 26)*C( 29) + R( 63)*C( 23)*C( 29)
     *     + R( 64)*C( 18)*C( 29)
C
110   CONTINUE
      DO 120 J = 1,NK
      YDOT(J+KS) = RATE(J)
120   CONTINUE
C
130   CONTINUE
      RETURN
      END



      SUBROUTINE STEADY(Y,N)
C
C     THIS VERSION OF STEADY COMPUTES THE STEADY STATE CONCENTRATIONS
C     OF O AND OH FOR THE ERT PHOTOCHEMICAL MECHANISM   (39X64) 1.31.79
C
      COMMON /CHEM1/ NOSTAT, NOSTM1, NOREAC, NOSPEC, NSTDY, NK
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),         RATKON(65),
     1              R(65) ,             RATEV(4,5),         QRATE,
     2              NVRATE,             LOCVRT(4)
      DIMENSION Y(1)
      DATA YES /3HYES /
C
      DO 50 K = 1,NOSTAT
      IF(QRATE.EQ.YES)  CALL RATEHI(K)
      J = (K-1)*NK
C     ---------   O   ----------
      FORM = R(1)*Y(2+J)
      DSTROY = R(2) + R(23)*Y(8+J) + R(24)*Y(9+J)
      CONIN(32,K) = FORM/DSTROY
C     ------   OH   ------
      FORM = R(6)*Y(4+J) + R(10)*Y(1+J)*Y(22+J) +R(55)*Y(15+J)
      DSTROY = R(7)*Y(1+J) + R(8)*Y(2+J) + R(9)*Y(5+J) + R(18)*Y(8+J)
     1  +R(19)*Y(9+J) + R(35)*Y(10+J) + R(42)*Y(7+J) + R(48)*Y(6+J)
     2  + R(52)*Y(14+J) + (R(50)+R(51)+R(57))*Y(12+J)
      CONIN(31,K) = FORM/DSTROY
50    CONTINUE
      RETURN
      END


      SUBROUTINE TEMPR(IT,TIME,T)
C
C     DETERMINES TEMPERATURE DEPENDENT REACTION RATES
C
      DIMENSION T(1)
      COMMON/CHEM2/ CONIN(40,5),        WTMOLE(40),        RATKON(65),
     1              RATEFF(65),         RATEV(4,5),        QRATE,
     2              NVRATE,             LOCVRT(4)
      RT = 1.986*T(IT)
C     HNO4 DISSOCIATION RATE
      RATKON(22) = 7.8E15 * EXP(-20700./RT)
      RATEFF(22) = RATKON(22)
C     N2O5 DISSOCIATION RATE
      RATKON(17) = 60.*5.7E14*EXP(-21051./RT)
      RATEFF(17) = RATKON(17)
C     PAN DISSOCIATION RATE
      RATKON(46) = 1.17E17 * EXP(-26910./RT)
      RATEFF(46) = RATKON(46)
      WRITE(10,1) TIME, T(IT), RATKON(17), RATKON(22), RATKON(46)
      RETURN
1     FORMAT(1H0, 1X,40HTEMPERATURE DEPENDENT RATES  TIME =
     I  ,F8.2,5X,6HTEMP =,F8.2,5X,7HRATES = ,3E10.3)
      END


      SUBROUTINE UNMIXR(KSTN)
C
C     DETERMINES RATE CONSTANTS AS A FUNCTION OF ELEVATION (MESH POINT)
C     FOR REACTIONS INVOLVING ALKE (HIGHER OLEFINS).
C
C     THIS ROUTINE IS SPECIFIC TO THE ERT PHOTOCHEMICAL MECHANISM
C     39 SPECIES X 64 REACTIONS   1.31.79
C
      COMMON /CHEM2/ CONIN(40,5),WTMOLE(40),RATKON(65),RATEFF(65),
     X  RATEV(4,5),QRATE,NVRATE,LOCVRT(4)
C
      IF(KSTN.GT.2)  GO TO 10
C          ---   50/50  -------
      RATEFF(18) = 5.8E+4
      RATEFF(23) = 1.46E+4
      RATEFF(25) = 4.58E-2
      RETURN
10    IF(KSTN.GT.4)  GO TO 20
C         ----  75/25   ------
      RATEFF(18) = 4.75E+4
      RATEFF(23) =    9.9E+3
      RATEFF(25) =  2.37E-2
      RETURN
20    CONTINUE
C          --  100/0   ---
      RATEFF(18) = 3.7E+4
      RATEFF(23) = 5.2E+3
      RETURN
      END


      SUBROUTINE UPFLX1(TIME,J)
C
C        *UPFLX1* UPDATES AREA SOURCE EMISSION FLUXES
C
C         THIS ROUTINE HAS BEEN MODIFIED TO PARTITION A SINGLE
C         ALDEHYDE FLUX INTO 60% HCHO AND 40% RCHO.
C         OLEFIN FLUX INTO HALF C2H4 AND HALF ALKE
C
      COMMON/FLUXES/FLXIN( 7,200),  FLXTIM(200),   NFLUX
      COMMON/CHEM3/ ZEE(5),                DELZ(4),               HTCELL(6),
     1              TDELZ(2) ,             DFINIT(6),             SCALOH(4),
     2              DCOF(5),               FLXWAL(40),            FLXDGE(40),
     3              BCFLAG(40),            DPRATE(10),            DEPOWR(10),
     4              LOCDPF(10),            NDPFLX ,               SCALUP(4)
      COMMON /CHEM5/  REACT(20,55),SPEC(40),LOCFLX(10),
     1                NASFLX,FLXW1(30),FLXW2(30)
      INTEGER BCFLAG
      IF(NASFLX.LE.0)  RETURN
      WRITE(10,1)TIME
      JP1 =MIN0(J+1,NFLUX)
      DO 100 I = 1,NASFLX
      K = LOCFLX(I)
      FLXW1(K) = FLXIN(I,J)
      FLXW2(K) = FLXIN(I,JP1)
100   CONTINUE
C
C         SPLIT ALDEHYDES
C
      FLXW1(7) = 0.40*FLXW1(6)
      FLXW2(7) = 0.40*FLXW2(6)
      FLXW1(6) = 0.60*FLXW1(6)
      FLXW2(6) = 0.60*FLXW2(6)
C
C       SPLIT NOX
C
      FLXW1(2) = 0.10*FLXW1(1)
      FLXW2(2) = 0.10*FLXW2(1)
      FLXW1(1) = 0.90*FLXW1(1)
      FLXW2(1) = 0.90*FLXW2(1)
C
C         SPLIT OLEFINS
C
      FLXW1(9) = .5*FLXW1(8)
      FLXW2(9) = .5*FLXW2(8)
      FLXW1(8) = .5*FLXW1(8)
      FLXW2(8) = .5*FLXW2(8)
      DO 200 I = 1,NASFLX
      K = LOCFLX(I)
      WRITE(10,2) SPEC(K),FLXW1(K)
200   CONTINUE
      K = 7
      WRITE(10,2) SPEC(K),FLXW1(K)
      K = 2
      WRITE(10,2) SPEC(K),FLXN1(K)
      K = 9
      WRITE(10,2) SPEC(K), FLXN1(K)
 1    FORMAT(1H0,10X,38HAREA SOURCE FLUXES UPDATED AT TIME   ,F10.2)
 2    FORMAT(1H ,10X,A4,3H = ,2E15.5)
      RETURN
      END



      SUBROUTINE UPSORC(TIME,IPS,NOSTAT,SPEC)
C
C         UPDATES POINT SOURCE EMISSION RATES
C
      DIMENSION SPEC(1)
      COMMON /PS1/   TPASS(200),     PS(7,5,75),     FRACT(3),
     1               NPTSR,          NPSFLX,         LOCPSF(7)
      COMMON /PS2/  PSRATE(30,5), PSR1(30,5), PSR2(30,5),TLAST,UPDINT
C
      IF(NPSFLX.LE.0)  GO TO 5
      IF(IPS.LE.NPTSR)  GO TO 10
 5    CALL XMIT(-150,0.0,PSRATE)
      RETURN
 10   CONTINUE
      IPSP1 = MIN0(IPS+1,NPTSR)
      DO 20 J = 1,NOSTAT
      DO 20 I = 1,NPSFLX
      K = LOCPSF(I)
      PSR1(K,J) = PS(I,J,IPS)
      PSR2(K,J) = PS(I,J,IPSP1)
 20   CONTINUE
C
C         SPLIT ALDEHYDES, NOX, AND OLEFIN  EMISSIONS
C
      DO 30 J = 1,NOSTAT
      PSR1(7,J) = 0.40*PSR1(6,J)
      PSR2(7,J) = 0.40*PSR2(6,J)
      PSR1(6,J) = 0.60*PSR1(6,J)
      PSR2(6,J) = 0.60*PSR2(6,J)
      PSR1(2,J) = 0.10*PSR1(1,J)
      PSR2(2,J) = 0.10*PSR2(1,J)
      PSR1(1,J) = 0.90*PSR1(1,J)
      PSR2(1,J) = 0.90*PSR2(1,J)
      PSR1(9,J) = .5*PSR1(8,J)
      PSR2(9,J) = .5*PSR2(8,J)
      PSR1(8,J) = .5*PSR1(8,J)
      PSR2(8,J) = .5*PSR2(8,J)
 30   CONTINUE
C
C       WRITE UPDATES ON TAPE10
C
      WRITE(10,1)  TIME
      DO 40 K = 1,NPSFLX
      ID = LOCPSF(K)
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)
 40   CONTINUE
      ID = 2
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)
      ID = 7                                                            NKM 1705
      WRITE(10,2) SPEC(ID),(PSR1(ID,J),J=1,NOSTAT)                      NKM 1706
      ID = 9                                                            NKM 1707
      WRITE(10,2) SPEC(ID), (PSR1(ID,J),J=1,NOSTAT)                     NKM 1708
      RETURN                                                            NKM 1709
    1 FORMAT(1H0,10X,37HPOINT SOURCE RATES UPDATED AT TIME =  ,F8.2)    NKM 1710
    2 FORMAT(1H ,10X,A4,4H =    ,5E12.3)                                NKM 1711
      END                                                               NKM 1712
```

Office of Research and Development
Environmental Research Information Center
Cincinnati, Ohio 45268

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE $300
AN EQUAL OPPORTUNITY EMPLOYER

POSTAGE AND FEES PAID
U.S. ENVIRONMENTAL PROTECTION AGENCY
EPA 335

U.S.MAIL

If your address is incorrect, please change on the above label tear off, and return to the above address.
If you do not desire to continue receiving these technical reports, CHECK HERE ☐*, tear off label, and return it to the above address.*

EPA-600/8-79-015b

B102/9